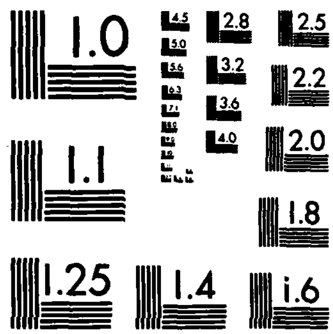
AD-A189 513 UNIT LEVEL WRSK (WAR READINESS SPARES KIT) ASSESSMENT 1/2
AND SORTIE GENERATI (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI T P LEWIS
UNCLASSIFIED DEC 87 AFIT/GOR/ENS/87D-9 F/G 15/5 NL



AFIT/GOR/ENS/87D-9

UNIT LEVEL WRSK ASSESSMENT

AND SORTIE GENERATION SIMULATION MODEL

THESIS

Theodore P. Lewis
Captain, USAF
AFIT/GOR/ENS/87D-9

AFIT/GOR/ENS/87D-9

UNIT LEVEL WRSK ASSESSMENT
AND SORTIE GENERATION SIMULATION MODEL

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Operations Research

Theodore P. Lewis, B.S.
Captain, USAF

December 1987

Approved for public release; distribution unlimited

Preface

HQ TAC/LGY uses Dyna-METRIC as a WRSK assessment tool but they have expressed a need for a more flexible model that is capable of running on a microcomputer. The purpose of this thesis work was to develop a model to emulate and extend the Dyna-METRIC modeling capability.

This model provides the initial emulation capability and some extended capabilities such as the ability to restrict maintenance and to have flexible scheduling. Areas for future work include more flexible input and output formats and the use of variance reduction techniques to reduce the number of simulation runs necessary.

In an effort of this magnitude, credit rarely rests with just the author. Therefore, I would like to thank my advisor Maj. Phillip Miller, whose time, wisdom, and patience have kept my thesis train from derailing. I am also grateful to Capt. Richard Mabe and Capt. Michael Budde for teaching me the secrets of Dyna-METRIC. To my typist and fiancee, Marcia Rossow, thank-you for your patience and love through a difficult time and for retyping Chapter 4 when the computer broke. Finally, to God, "For from Him and through Him and to Him are all things. To Him be the glory forever. Amen (27:Romans 11:36)."

Theodore P. Lewis

Table of Contents

List of Figures

List of Tables

AFIT/GOR/ENS/87D-9

Abstract

HQ TAC/LGY uses Dyna-METRIC as a WRSK assessment tool but they have expressed a need for a more flexible model that is capable of running on a microcomputer. For example, Dyna-METRIC has a number of limiting assumptions such as assuming unlimited maintenance capacity. The purpose of this thesis work was to develop a model to emulate and extend the Dyna-METRIC modeling capability.

To begin this research a simulation package had to be chosen. Microcomputer simulation languages were compared and SLAM II PC was selected because of its price, portability, widespread acceptance as a simulation language, and the availability of the software.

Another area of concern was Dyna-METRIC's use of the exponential distribution to model repair times. Questions have arisen as to whether this is a reasonable assumption or whether the lognormal distribution provides a better fit. The sample repair times were taken from a TAC exercise called Coronet Warrior. The results were inconclusive due, primarily, to the small sample sizes. Testing of the research model centered around two data sets. The first was

provided by HQ TAC/LGY and the second came from the TAC Coronet Warrior exercise. The outputs of interest are sorties per day and number of fully mission-capable aircraft available per day. Each data set was used with the research and Dyna-METRIC models. The outputs were then compared by day and type. A hypothesis test of the differences was performed. The differences were not found to be statistically different from zero. Therefore, the research model provides a reasonable emulation of the Dyna-METRIC model with respect to the outputs of interest. Future research is recommended in input and output formats and in variance reduction techniques to reduce the number of simulation runs necessary.

UNIT LEVEL WRSK ASSESSMENT AND SORTIE GENERATION SIMULATION MODEL

1. Introduction

Background

Tactical Air Command, like any Air Force unit, has a need to manage resources efficiently and assess the capabilities of these resources. As a part of this management effort, the Tactical Resources Analysis office at Tactical Air Command headquarters is often tasked with analyzing the efficiency of Tactical Air Command's resource allocation. One area of concern is the management and assessment of the War Readiness Spares Kit (WRSK) recoverable spares (30). The War Readiness Spares Kit is the Air Force standard set of spares. This kit allows an Air Force unit to fight for 30 days without external resupply. Recoverable spares are those parts in the aircraft which can be repaired and used again. To model this situation, Tactical Air Command Resources Analysis office uses Dynamic Multi-Echelon Technique for Recoverable Item Control (Dyna-METRIC). Dyna-METRIC is an analytical model which runs on a mainframe computer at Tactical Air Command headquarters (30). Although Dyna-METRIC is a flexible tool, it has a number of limitations due to its analytical nature, which have restricted its effectiveness in realistically predicting sortie generation rates (number of times an aircraft can fly per day) and WRSK capability at the squadron level. This model assumes, for example, that unlimited maintenance personnel and test equipment are available, that all parts are mission-essential, and that all parts fail at the Air Force wide failure rate (15:1). Due to these shortfalls Tactical Air Command wants a more flexible model which can emulate Dyna-METRIC for validation purposes and yet can also be expanded to address the areas where Dyna-METRIC is limited (15:2).

Specific Problem

The specific problem is that TAC lacks a flexible tool to use in evaluating current WRSK needs and capabilities because the current model has restrictive assumptions.

Research Objective

Develop a tool that emulates the Dyna-METRIC resource management and spares capability assessment model and is also capable of addressing Dyna-METRIC shortfalls, yet still runs on a microcomputer.

Sub-objectives

The sub-objectives are to:

- Investigate the limitations of Dyna-METRIC.

- Evaluate the solution techniques available and choose the most suitable one for Tactical Air Command's resource management needs.

- Develop a model that emulates Dyna-METRIC at the squadron level, yet is capable of being expanded to address selected Dyna-METRIC limitations to include the following:

  -- An aircraft scheduling module that includes daily flight schedules.

  -- Flightline and intermediate level maintenance constraints.

- Develop an interface for use with the Dyna-METRIC input data base.

-Provide an appropriate output to the model user.

-Provide user documentation for the model.

The rest of this document gives the steps taken to meet these objectives. Chapter Two provides a review of the literature pertinent to this research, while Chapter Three discusses the approach or methodology used. Chapter Four reports the analysis and results of testing. Chapter Five will draw conclusions and suggest areas for future research. The appendices will include data tables, model flow diagrams, model coding, and model output.

2. Literature Review

Models

A model is an abstraction or idealization of a real world situation (29:211). As with any imitation, it provides an incomplete representation of the real thing, says Specht. Pritsker gives four reasons for simulation models:

> ... as explanatory devices to define a system or problem; as analysis vehicles to determine critical elements, components, and issues; as design assessors to synthesize and evaluate proposed solutions; as predictors to forecast and aid in planning future developments [23:1].

The purpose of this research model falls in the second and fourth categories. Specht (29:212) and Pritsker (23:11-12) both agree that the scope and complexity of the model are dependent on the problem to be solved. This implies that problem definition is critical in developing an appropriate model. Both Pritsker (23:11) and Specht (29:218) also agree that deciding what is relevant in the model and the criteria or objectives of the analysis are important. Pritsker provides a ten-step evolutionary process which he recommends for developing a simulation model. The steps are:

1) problem formulation

2) model building

3) data acquisition

4) model translation

5) verification

6) validation

7) strategic and tactical planning

8) experimentation

9) analysis of results

10) implementation and documentation (23:11).

These are similar to the steps given by Lee, Moore, and Taylor (18:463). Pritsker points out that no simulation project is complete until it is used in the decision making process (23:13) but Ellsberg, a Rand staff member, also warns that the use of models

> ... will not eliminate uncertainty or insure correctness; will not forsee all major problems, goals, contingencies, and alternatives; will not eliminate the necessity of judgement or the effect of bias or preconception [29:226-7].

Ellsberg does agree, however, that it should enhance the decision process (29:227).

Simulation

Dalkey says that simulation is one of the most important tools of the military analyst because military conflict involves a complex interaction of many items including weapons, strategies, and time. He also points out that it is often the only method which can provide accurate and reproducible results (8:241). Lee, Moore, and Taylor state that the primary reason for simulation is that many real world problems do not lend themselves to mathematical modeling and solution to optimality because of stochastic relationships or problem complexity (18:461). They also indicate that one reason for the popularity of simulation is its flexibility, which allows fitting the model to the problem rather than the problem to the model (18:491). A number of examples of simulation applications are given by Pritsker (23:Chap 4), Lee, Moore, and Taylor (18:489), and Dalkey (8:248). The applications touch almost every aspect of the business and military world.

Before using simulation, however, the advantages and disadvantages of simulation should be weighed. The advantages given by Dalkey include the ability to handle complex systems and the ability to break down the system into smaller, simpler sections, creating more easily understood situations for the decision maker. Also, the assumptions are usually clearly stated and results can be duplicated. Simulation provides a logical framework and often can be used as a self-check of the model assumptions (8:250). Pritsker would also add that simulation allows drawing inferences

> ... without building them, if they are only proposed systems; without disturbing them if they are operating systems that are costly or unsafe to experiment with; without destroying them, if the object of an experiment is to determine their limits of stress [23:6].

Dalkey also gives some disadvantages of simulation. For example, simulation is often a slow and cumbersome method for solving a problem. The models can be difficult to change and restricted to only a few situations. The results are often treated with undue respect and there is also a tendency for the user to treat the model as a black box with no understanding as to what is going on inside the model. It is also difficult to simulate a commander's decisions (8:251). In choosing to use simulation both the strengths and weaknesses should be kept in mind.

War Readiness Spares Kit (WRSK)

Basic military doctrine centers on three subjects. These areas are the weapon system, the supply or logistics system, and manpower. Manpower and logistics are the independent variables, while the weapon system is the dependent variable (10:1). To stress the importance of logistics, General Eisenhower said, "You will not find it difficult,...,to prove that battles, campaigns, and even wars have been won or lost primarily because of logistics (14:XII)." During peacetime operations, supplies are kept close to where they will be used but, when hostilities arise, units can be deployed worldwide outside of established supply chains. Due to the need to plan for these contingencies, the military has developed the War

Reserve Material Program. This program is designed to support deployed operating units and relies on prepositioning of materials based on preplanned programs and schedules (32:14-3). In the event of hostilities the War Reserve Material (WRM) stock is additional equipment held in reserve which supplements normal peacetime operating stocks until industrial production can sustain combat requirements. It includes spares, equipment, war consumables, and medical material designated as WRM by AFR 400-24 (36:1). The War Readiness Spares Kit (WRSK) is a part of the War Reserve Material program for units with aircraft, vehicles, communication systems and other appropriate systems. A WRSK is defined as an air-transportable kit of critical spare parts to provide sustained operations during wartime or contingency when normal supply channels are interrupted or fall short of demand. They are meant to sustain a unit for some specified period of time (usually 30 days) without external resupply (32:14-13).

The development of the War Reserve Material Program began at the end of World War II. The need for some war readiness capability was evident from America's lack of capability at the beginning of World War II and from the post-war political climate. In 1946 the original requirements for a readiness capability centered around the deployment of bulk supplies and equipment and the need for a 30 day unsupported maintenance capability. In 1948 the

Strategic Air Command created an airborne kit which would allow aircraft operations for 30 days from forward bases without any logistics support (13:5).

Literature on war readiness material is scarce and focuses predominantly on how to deal with funding problems and lack of war readiness materials, rather than on the development of the proper war readiness material stock composition. This situation should be expected because the determination of logistics requirements has always been difficult. This is due in part to the constant changes in requirements to support contingencies (13:6).

The prima.y planning document for War Reserve Material is the War and Mobilization Plan (WMP). In time of war, support needs are met by the peacetime operating stock and the War Reserve Materials stock. This reserve stock provides additional equipment so that military units can go from low peacetime consumption rates to high wartime consumption rates. The basis for these changes in consumption rates is established in the DOD Defence Guidance (DG). Details on specific forces and scenarios used to determine requirements are in the WMP-1 (35:5). The policy and responsibilities for management of War Reserve Materials is contained in Air Force Regulation 400-24. Air Force Manual 67-1 provides further details on these responsibilities. The logistician faces a difficult problem in determining WRSK composition because of the need to predict demand under uncertain conditions. Also, his resources are limited because

> ... prepositioning and prestocking War Reserve Material in the range and scope necessary to cope with every possible combination of circumstances would bankrupt the country [13:4].

He must, therefore, budget his resources to provide the best mix of mobility and flexibility across the range of conflict (13:4).

Tactical Air Command uses War Readiness Spares Kits in order to accomplish this goal. These War Readiness Spares Kits represent the sole source of aircraft spare parts during the initial phases of conflict. Because of the need to deploy worldwide and operate initially as a logistically independent unit, these kits are prepositioned with the unit and are air-transportable (32:14-13). The kits are comprised of the following three elements:

--Enroute Support Team (EST)

--Initial Support Element (ISE)

--Tactical Support Element (TSE)

The Enroute Support Team (EST) includes the WRSK assets required to move the aircraft to the deployed site. The Initial Support Elements (ISE) are WRSK assets necessary to support operations during the first seven days of the deployment. Combined with the Enroute Support Team, they form the Leading Edge package and are built to support a deployed unit at WMP-5 surge rates (33:9-10). The Tactical Support Element is the remaining WRSK necessary to sustain a deployment from day eight to day thirty. TAC uses the following criteria to measure WRSK program effectiveness:

A) Inventory accuracy

B) Control of shelf life/functional check items

C) Accuracy of WRSK authorizations

D) Receiving and storing of assets

E) Issuing Assets

F) Appearance of the WRSK. (33:3)

War Reserve Material is crucial to USAF war planning, thus supervision, control, and use of this material falls to the storing command. This material should be segregated from other base supply stocks. Although a War Readiness Spares Kit (WRSK) may be used to bring an aircraft or end items authorized war reserve material support back to operational condition, it should not be considered a source of continuous supply because of its wartime mission (32:14-5). Selection criteria for war reserves reside in AFR 400-24. They include such things as:

- items essential for combat forces:
  - -- to destroy an enemy's capacity to continue fighting.
  - -- to give battlefield protection of personnel.
  - -- to detect, locate, and maintain surveillance of the enemy.
  - -- to communicate under war conditions.

- items essential for operational effectiveness of combat support units.
- items essential to effective weapon or equipment operation.
- items essential for sudden mobilization or deployment.
- items designated as operational rations (35:4).

TAC units are tasked to use the dedicated crew chief program for aircraft WRSK management and each crew chief is responsible for managing all facets of his WRSK kit (33:2-3). The Non-Commissioned Officers-In-Charge of the War Readiness Section in the Aircraft Generation Parts Store or the Non-Commissioned Officer-In-Charge of the Tactical Air Control System Material Control are the WRSK/BLSS (Base Level Self-sufficiency Spares) custodians (33:7). Resupply of the War Readiness Spares Kit comes from two sources. The first is from repair in the field and the second is from resupply channels. Items that are removed on the flight line are sent to field level maintenance for repair and then returned to the WRSK. Items broken beyond the unit's repair capability are sent to a higher level and a new part is ordered (10:23).

Dyna-METRIC Model

Demmy and Hobbs give some background to the Dynamic Multi-Echelon Technique for Recoverable Item Control (Dyna-METRIC) model. Work began on optimization techniques for stationary, multi-echelon, multi-indenture inventory/repair systems in 1966 by Feeney and Sherbrooke. This model calculated backorder levels for depot and base stock levels and provided the basis for recoverable item requirements in the Air Force (9:14).

Pyles gives the need for the Dyna-METRIC model development by stating, "A technique that merely assesses alternate logistics decisions would be inadequate" (24:1). Because of the large number of items managed, there is a need for a forecasting technique which can identify spares shortfalls. Pyles emphasizes that previous models always looked at noncombat measures such as part backorders while Dyna-METRIC assesses how alternate part support processes and resources will impact combat capability (24:1). Pyles and Tripp also stress the Air Force Logistics Command's interest in developing a tie between support decisions and combat effectiveness. Two important aircraft measures of effectiveness are the number of fully mission-capable aircraft and the number of sorties flown based on war plans or peacetime flying goals (25:18).

The original METRIC model was based on a steady state inventory system with constant average demand rates and constant average service rates (16:1). The demand rates in certain situations such as wartime or peacetime exercises are not steady state (16:1). Hillestad and Carrillo in their Rand note give a development of the mathematics necessary to handle this transitory state (16). They also stress the importance of three assumptions inherent in this model. These assumptions are "sufficient slack service capacity, independence of the service and demand process and poisson arrivals (16:23)." If these assumptions are not valid for a given situation, then the results may no longer be valid (16:32).

Demmy and Hobbs describe Dyna-METRIC in the following manner. Dyna-METRIC has a three-layer inventory and repair system consisting of a depot, a flexible number of centralized intermediate repair facilities, and the main operating bases. The depot portion of the model is limited. The model enables the user to either include or exclude the intermediate repair stations depending on his or her environment. The internal flow of parts within the model depends on the repair capability assigned to each level of repair. For example, main operating bases usually have limited repair capability. An item that fails is removed from the aircraft and replaced with a serviceable (operable) spare. The item that is removed and replaced on the flight line is called a line replaceable unit (LRU). These units are then sent to base supply. Based on the severity of the problem and the availability of a centralized intermediate repair facility, the part will either be repaired at the base, sent to a centralized repair facility, or sent to the depot. Base resupply within the model is accomplished in a similar manner. Depending on where the item is repaired, the part will flow back to the base from that station (9:14-15). Two things should be noted. First, where the part is repaired has a significant effect on resupply time and second, resupply is always done on a one-for-one exchange between the base and the repair station (9:15).

According to Gage and Ogan, the model has a number of limitations which should be kept in mind when considering possible operating scenerios and when using the results obtained from the model (12:23-24). First, the actual number of sorties flown can never be more than the number demanded. This may give the false impression that a unit can do no better than the expected sortie rate, which is obviously misleading. Secondly, the number of sorties demanded sets the consumption of spares rather than the actual number of sorties flown. This limitation implies that spares are used as though all missions are being flown even when all are not being flown. Next, the model does not discriminate between grounded (non-mission capable) aircraft and aircraft which are capable of performing some but not all of their missions (partially mission capable). The model counts all broken aircraft as non-mission capable, distorting the analysis. Another limitation is the fact that ample repair facilities are always assumed to be available to perform repairs. This limitation implies that there is never a backlog in the maintenance shop and thus, there is never a delay in a repair because of a sudden increase in broken parts. This can lead to a better repair time than actually exists, an underestimate of spares needs, and therefore, a shortage of parts. The model also assumes that repair and demand processes are independent. The repair process is a first-in, first-out line regardless of the importance or need of the flight line. Therefore, if a part is really needed on the flight line, it can not be rushed or placed at the front of the line. The failure rate of parts is assumed to only be affected by the number of hours flown. The depot is considered to be an infinite source of parts and therefore, parts will never be out of stock. This assumption implies that the depot is always able to issue parts immediately. Finally, the model assumes that the centralized intermediate repair facilities distribute spares based on cumulative flying hours and no other basis. This limits the flexibility of meeting unique base needs or shortages within the model and therefore is a pessimistic assumption (12:23-24). Despite these limitations Dyna-METRIC is still a powerful tool as Demmy and Hobbs point out.

> The usefulness of Dyna-METRIC appears to be in its ability to evalute the dynamic response of a logistics system when operated in various configurations and under various policy alternatives relating to transportation, provisioning, maintenance, and deployment. It provides performance measures at both the item and system levels which increases its utility as a tool for aiding decision making [9:17].

Microcomputers

One of TAC's requirements is that the simulation model be available on a microcomputer (15). Arthur, Frendewey, Ghandforoush, and Rees did an extensive review of over 20 microcomputer simulation packages (3). They point out that the new widespread availability of simulation packages for the microcomputer is making simulation much more affordable. However, they also state that if the user has reasonable access to a mainframe, the mainframe is almost always preferable, primarily for speed reasons (3:167-168). The review had two purposes. The first was to provide a good list of simulation packages available for the microcomputer and the second was to further investigate several of the simulation packages which have mainframe counterparts (3:168). They looked only at how well the language was implemented on the microcomputer and not the inherent quality of the language. They also did not identify a best overall package (3:168).

Table 10 found in Appendix A gives a summary of the simulation packages in this article (3:169-172). The four microcomputer implementations that Arthur and the others considered in detail were GPSS/PC, SIMAN, SLAM II PC, and PC SIMSCRIPT II.5. They believe that these are the most commonly used mainframe simulation languages. They also feel these four microcomputer versions are quality implementations of their mainframe counterparts (3:173).

Table 1, taken from the article, gives a list of the operating requirements for each microcomputer simulation language.

Table 1 Operating Requirements
(Reprinted from 3:174)

	GPSS	SIMAN	SLAM	SIMSCRIPT
Type of computer	IBM PC or compatibles	IBM PC or compatibles	IBM PC or compatibles	One based on the Intel 8086/88, 80186/88, or 80286 chip family
Memory	256K	320K	320K	320K (640K preferred)
Disk drives	1 Floppy	2 Floppy; Hard disk recommended	2 Floppy*; Hard disk recommended	5 megabytes hard disk
Operating system	DOS 1.1, 2.0	MS-DOS 2.XX	MS-DOS 2.XX	PC-DOS & corresponding versions of MS-DOS
FORTRAN compiler required	No	Yes†	Yes‡	No
Numeric processor	No	Recommended	Recommended	Required

*Only one floppy drive required for network models.
†Not required for block diagram models.
‡Not required for network models.

Table 2, also taken from the article, gives a concise comparison of the four languages.

Table 2 Language Comparison
(Reprinted from 3:174)

	GPSS	SIMAN	SLAM	SIMSCRIPT
1. Modeling approach				
Event	No	Yes	Yes	Yes
Process	Yes	Yes	Yes	Yes
Continuous	No	Yes	Yes	No
2. Pseudorandom numbers Generation method	PMMLCG*	PMMLCG	PMMLCG	PMMLCG (implementation dependent)
Can user supply a generator?	Yes	Yes	Yes	Yes
3. Support				
Built-in random sampling distributors	None	Excellent	Excellent	Excellent
Debugging aids	Excellent	Good	Good	Excellent
Editing	Excellent	None	None	Excellent
4. Computer runtime				
Micro Preparation Phase	N/A	39 sec	57 sec	20 min compilation
Simulation Phase	142 sec	63 sec	101 sec	482 sec
Mainframe	0.12 sec	≃ 1 sec	≃ 1 sec	≃ 1 sec
5. Documentation				
Language	Excellent	Excellent	Excellent	Excellent
User's Manual	Good	Fair	Excellent	Poor
6. Ease of getting model to run	Good	Fair	Good	Good
7. Ease of modifying the model	Excellent	Good	Good	Excellent/Good
8. Output analysis provided	Excellent	Excellent	Excellent	N/A
9. Report generation capabilities provided	No	No	No	Yes
10. Portable to mainframe	90%	Yes	Yes	Yes
11. Cost (non-educators/educators)	$900/900	$1500/200	$975/200	$24000/250

*PMMLCG = prime modulus multiplicative linear congruential generator.

The authors stress that the responses in the table are subjective and based on their wide experience and not on any large sample questionnaire of users. The authors also indicate that they only looked at basic simulation packages and not any enhancements which might be available (3:173).

In reviewing Table 2, the authors pointed out a number of items which are important to this thesis. First of all, GPSS is limited to process-interaction type models and SIMSCRIPT can not model continuous systems. All packages are considered at least good in support issues and excellent in language documentation. User manual documentation, however, varies significantly. When considering run times, the authors stress that the reader should not compare times between packages because runtime is very dependent on what is being modeled. The most important thing to note is the increase in runtime between the mainframe and the microcomputer versions. The authors rated all languages at least good when trying to modify or run a model except SIMAN, which was somewhat more difficult to run. Excellent output analysis was available with all packages except SIMSCRIPT II.5. All packages were completely portable to a mainframe except GPSS which is 90% portable. Finally, cost was considered. Because of TAC's requirement for multiple site implementation of this model (30), SIMSCRIPT II.5 quickly becomes cost prohibitive at 24,000 dollars a copy. The other three packages are comparably priced at about 1000 dollars and should be considered further.

Several other languages also exist and should be considered. Vasudev and Biles, seeing the need for a microcomputer simulation language, are pushing for the development of a mainframe-like, general purpose simulation language and recommend Pascal Simulation Modeling and Analysis Program (PASMAP) as a step toward fulfilling that need. The Pascal simulation package combines discrete-event, continous, and state event simulation. It also allows user input and output interaction (14:30-31). However, there is one critical shortfall of this model for this thesis. This package is still under development and not available for use. Another simulation package to consider is UCSD Pascal. O'Keefe and Davies give a number of reasons to use UCSD Pascal (22). First, UCSD Pascal tends to be self-documenting and easy to learn. Also, because of the Pascal implementation, the size of a simulation program is not an effective barrier (22:25). There is one important problem. UCSD Pascal is only available on Apple II computers and not the Zenith computers (30).

Verification and Validation

Whenever a model is built, the question usually arises, "Does the model represent reality ?" In addition, this question is asked, "Why not just use historical data for validation ?" Dalkey says the modeler usually can not guarantee a model of reality. Use of historical data is often difficult because it is not available especially in a scenario similar to the model. Even if historical data is available, the results might disagree and the model still be valid. This occurs because of the effects of chance, the level of detail in the model, and the commander's decisions on historical outcomes. A major determinant in war is a commander's decisions and presently there is no adequate way to model this process (8:251-2). Specht also has something to say on these matters. He points out that often, because of the lack of data, the builder can't adequately test the model but hopefully the model can answer the following:

> - Can the model describe correctly and clearly the known facts and situations ?
> - Do the results remain constant and plausible, when the principal parameters involved are varied ?
> - Can the model handle special cases in which there is some indication as to what the outcome should be ?
> - Can the model assign causes to known effects ? (29:220)

Shannon says it is impossible to prove that any simulation is a correct or true model of reality but stresses that we are seldom concerned with proving truth. Instead, he promotes validating the insights gained from the simulation. This, he stresses, is the utility of a model, not truth

(28:29). Specht adds that the modeler should not be concerned that the model doesn't look like the real thing or that it doesn't model all of reality. He indicates that the important point is that the output of the model is reasonable and valid, adding that several models may even exist for the same reality because model structure depends on the object modeled, the questions asked, and the decisions affected (29:212).

Since the ultimate goal of any model is to aid decision making, the builder would like to test for the correctness and relevance of the results. Since this is not always possible, often the best thing to hope for is truthfulness (12:212). Further, no matter how the modeler strives to maintain a scientific inquiry or to follow scientific methods, military systems analysis is not an exact science. Although it may appear purely rational, coldly objective, and analytical, don't be fooled. Human judgement is used for: 1) designing the analysis 2) choosing alternatives, 3) choosing relevant factors, 4) picking interrelationships, 5) gathering data and 6) analyzing and interpreting results. Thus, Quade cautions both analyst and decision maker to avoid errors that bias results (26:363). Specht gives a similar warning;

> This fact - that judgement and intuition and guesswork are embedded in a model - should be remembered when we examine the results that come, with high precision, from a model [29:220].

Verification and validation should be performed whenever a model is built. There are a number of definitions available for these terms. Shannon says that validation is "Increasing to an acceptable level the confidence that an inference drawn from the model about the real system will be correct (28:23)." Pritsker says validation is the "process of establishing that a desired accuracy or correspondence exists between the simulation model and the real system (23:11)." While verification is the "process of establishing that the computer program executes as intended (23:11)." Fishman and Kurat divide model evaluation into three categories; 1) Verification - model behaves as the modeler intended, 2) Validation - testing the model vs. the real world, 3) Problem Analysis - model capable of drawing significant inferences from the simulation (28:20).

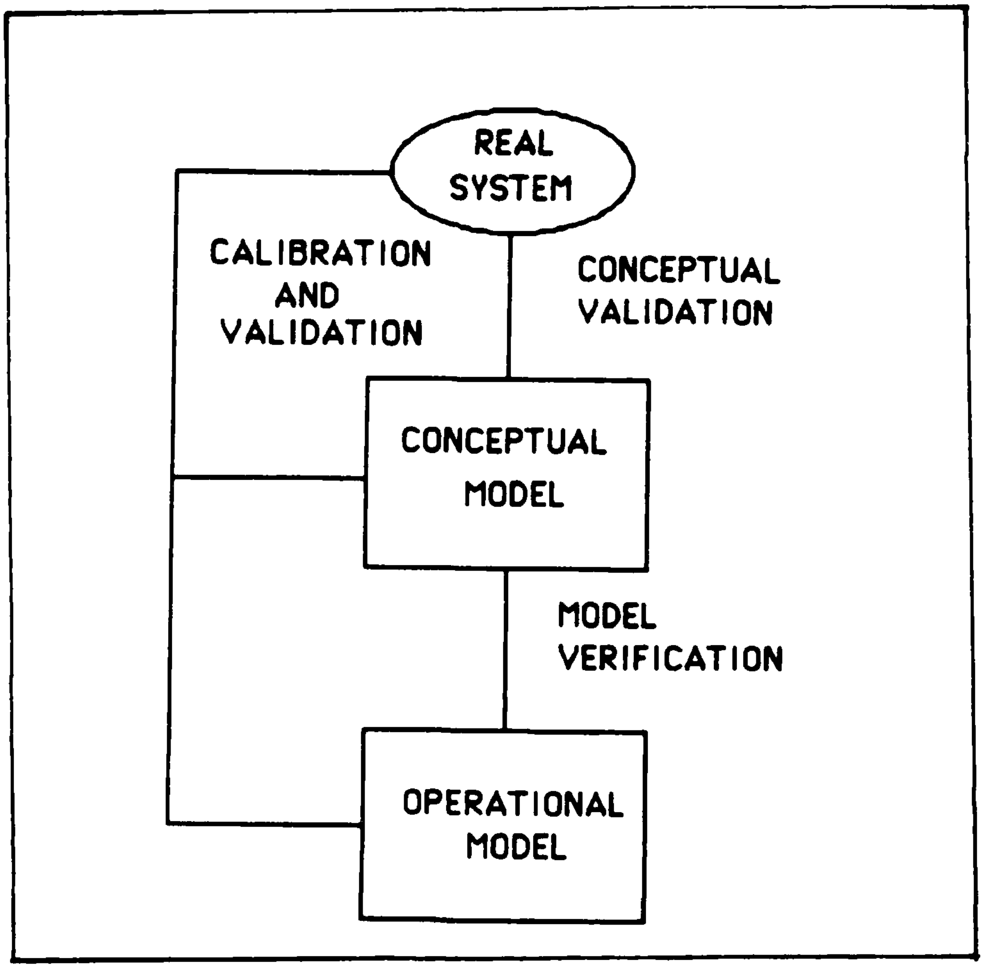


Figure 1. Verification and Validation Concepts (19)

The preceding figure gives a conceptual picture of the verification and validation process.

Shannon gives three tests that could be used: 1) testing the face value of the model (experts in the field look at the structure and at the input/output transformation to see if they can tell the difference between the real world and model data), 2) testing model assumptions and 3) testing input/output transformations to the real world. Two and three above can use a variety of statistical tests including tests of means and variances, analysis of variance, regression analysis, factor analysis, spectral analysis, autocorrelation, chi-square tests, and non-parametric tests. But Shannon warns the tester to remember that each method involves a set of assumptions which must be true for the analysis to be correct (28:29). Hillier and Lieberman also recommend using standard statistical tests if the data exists for comparisons. If data doesn't exist they recommend face value tests but add the following ideas:

1) Use a field test to collect data, if possible.
2) Use experts to look at the sensitivity of the model in a variety of scenarios.

The problem with the first method is that it is often expensive and time consuming and the system may not be available or easily duplicated. The second method can be valuable because, even if conclusions about a single scenario are weak, some things about changes in results between scenarios may be important (17:809-810). Lee, Moore, and Taylor believe that the major difficulty is validation of the model. First, they recommend checking the algorithm to be sure it is correctly coded to avoid the garbage in - garbage out problem. Then they recommend running the model for short periods of time and comparing these results with hand-calculated ones. They also recommend breaking the model into modules for testing. To aid initial testing they recommend simplifying complex relationships such as changing stochastic relationships to deterministic ones. Finally, recognizing the difficulties associated with this, they recommend comparing the model with the real world. They also warn the tester of problems with the statistical assumptions necessary and of problems with the covariance that sometimes exists in simulation models over time (18:486). Starting conditions should also be checked and important steady state conditions, if they exist, should be analyzed to prevent output bias (18:487). Although most authors recommend using past performance of the real world situation as a guide, Pritsker warns the tester to remember that past performance is only one sample point and not the exact answer (23:12). Testing suffers from three common research problems: 1) small sample size due to high cost, 2) aggregated data, and 3) data with questionable validity (28:30).

Torn provides a method for easily modeling simulation designs and then validating them. This method uses Petri nets with extensions called simulation nets. Petri net theory is well established and has been in use since 1962 (31:71). Torn lists the following advantages to using simulation nets:

1) The tool is based on the widely used Petri nets.

2) Petri nets are well suited for describing asynchronous concurrent processes.

3) Petri nets are theoretically well founded.

4) The tool is simple and easy to learn.

5) Both top-down design and independent modeling (including validation) of different aspects is possible.

6) The tool is user-friendly, computer programming is avoided and the whole effort is devoted to modeling (31:71).

This simulation net method does not describe how the simulation should be performed but describes only the model itself. Because of this non-procedural method, results are calculated much more quickly than when using a method that requires programming (31:71). Finally, Torn discusses the advantages of simulation nets in validating simulation designs. He points out that this method is well-founded and especially built for describing parallel processes. The technique has only a few modeling pieces and therefore is easy to learn and use. Also, the model allows independent design of each module. Because of the rigid Petri net format, the modeler can be confident in his model flow and testing the model becomes easy and efficient. This is due, primarily, to the fact that direct use of the nets allows checking of the simulation without having to write a computer program first. Torn also suggests that more research should be done in this area (31:75).

Although simulation models are simplifications of their real world counterparts, they can still be extremely complex, difficult to work with, and expensive to use. Because the simulation model may still be complex, L. Friedman and H. Friedman suggest the use of a metamodel. This simpler model would be used to aid understanding of the more complex model. They provide several examples of fields in which metamodels have been used to help interpret more complex models. These metamodels are especially useful in performing sensitivity analysis and answering what-if questions because they don't use a computer; this can save time and money. One example of a metamodel is the well known regression model (11:144).

Since the metamodel is two levels from reality, the authors stress the importance of validating the metamodel both to the simulation model and to the real world. To validate the metamodel the authors give two techniques. The first involves randomly dividing the simulation data into thirds. The first two-thirds will be used to build a second regression model. The remaining third of the data is used to test this regression model. If the R-squared value (a measure of model fit) from the test data is small, the model does not explain the variation in the data very well and this indicates a lack of predictive ability to be a useful model (11:145). The second technique is called double cross-validation. The simulation data is split in half and a model is developed for each half of the data. Then the opposite half of the data is used to test each model. This provides four R-squared values. These values are then compared for consistency. Large variations in values imply that the metamodel is probably invalid. In addition, the coefficients of each model should also be compared. If large differences exist between the models especially with respect to the coefficient's signs, then the metamodel may be inconsistent (11:145-6). Finally, the authors stress the need to test the predictive ability of the metamodel against the real world because this is the final test of any model. The metamodel should not only fit the simulation model but the real world too (11:146).

In conclusion, Hillier and Lieberman stress that the most important thing is to convince the decision maker of the model's validity so that he will use the results to aid his decisions (16:809).

> The question of validation is thus two-faced: determining that the model behaves in the same fashion as the real life system; validating that inferences drawn from the experiments with the model are valid or correct. In concept, both these points resolve themselves to the standard decision problem of balancing the cost of each action against the value of the increased information and the consequences of erroneous conclusions [28:30].

Other Models

Several models which address aircraft maintenance and logistics problems are available. This, however, does not negate the importance of this thesis effort because models of the same reality often occur as a result of the object modeled, questions asked, and the decisions affected (29:212). The following is a short description of some of the models available. The Expected Value-Based Logistics Capability Assessment Model (ELCAM) is a model being built at the Air Force Logistics Management Center for assessing logistics capabilities. It is not meant to replace any of the current mainframe models such as the Logistics Composite Model (LCOM), Theatre Simulation of Airbase Resources (TSAR), TSAR Inputs using Airbase Damage Assessment (TSARINA), or Dyna-METRIC models, but as a supplement to the current methods being used. Considered a first order approximation, the documentation clearly states that the purpose of the model is to give the logistics manager an easy-to-use tool for trade-off and sensitivity analysis. The model is not meant to be an in-depth logistics analysis tool (2:1). The model is not a simulation but an expected-value-based model. As such, it lacks some of the dynamic nature of a simulation (2:10). The key to the model is determining the expected value of downtime for an aircraft. If the model can accurately do this, the theory is supported (2:3). Although the model is not a Dyna-METRIC emulation, it does use some mathematic techniques from Dyna-METRIC such as those used to calculate the number of pipeline spares (2:5). One model drawback is that it also bases failures on scheduled flying hours rather than actual flying hours, but work is being done to address this limitation (2:10). The model is far from its final form and has a number of limiting assumptions but it wasn't meant to be an in-depth analysis tool. It is written in Basic for the Z-100 or the Z-248. Testing with small sample data bases from TSAR has given promising results (2:7).

BDM has built a model known as the Availability/Readiness Model for the Personal Computer (ARM-PC) (4). The uses of the model include specification requirements evaluation, design influence, design assessment, and general management assistance. The model is a stochastic network model in SLAM II and runs on a micro-computer. It was built to simulate a generic aircraft operation and maintenance environment. Although maintenance is at the LRU-level, output is aggregated to the subsystem level. It is a generic aircraft logistics model owned by BDM and not meant as a Dyna-METRICS emulation package. It is meant primarily for assessing design change impacts and what-if analysis (4).

The Aircraft Subsystem Availability Model (ASAM), developed by Headquarters Air Force Operational Test and

Evaluation Center is designed to estimate an aircraft subsystem's performance (1). The model does this by estimating the subsystem availability in terms of mission-capable rate. Able to calculate the subsystem downtime at both the organizational and intermediate maintenance level, the model also estimates the impacts of manpower and resource constraints on the subsystem availability. The model supports both peacetime and wartime scenarios (1:2). The model, however, has a number of limitations. First, the model only looks at one subsystem and aggregates the rest of the aircraft. Also, the subsystem is limited to only 20 LRU's. Another problem is that the model cannot handle multiple LRU failures. The model has a fixed aircraft squadron size and a scenario of cne year of peacetime operation (20 sorties/day). No combat or weather losses are included. Maintenance times in the model use a lognormal distribution (1:2). The model is built in SLAM II and runs on a Z-248. HQ TAC has validated the model logic and input variables (1:6).

Coronet Warrior

Dyna-METRIC has been plagued by a problem common to many models (ie. validation). Because of a lack of supporting data, Dyna-METRIC has had problems gaining acceptance as a requirements builder. In an attempt to remedy this problem, HQ TAC has performed an exercise to provide data for a Dyna-METRIC validation. This exercise took place in the summer of 1987 using an F-15 squadron which was temporarily isolated from the rest of the base. The object of the test was to simulate a deployed unit with limited manpower, equipment and resources. The unit was tasked to fly realistic combat sortie rates for 30 days. Data was collected on everything that was done during the test. This data would later be used in the Dyna-METRIC model to see how well the model predicted reality. Then, if necessary, changes would be made to improve Dyna-METRICS. Because this thesis effort attempts to emulate the Dyna-METRIC model, the results of this exercise are important for comparison with the Dyna-METRIC results and for comparison to the real world. Several important discoveries were made during the test including:

1) Demand for parts was lower than predicted.
2) Parts were fixed faster and more reliably than predicted.
3) Personnel adapted and performed better than predicted. (7)

Figure 2 shows the number of demands predicted by Dyna-METRICS and the actual exercise demand. The Figure also shows the initial number of WRSK parts available and the number remaining at the end of the exercise, predicted and actual. The deployed unit contained only 71% of the standard Air Force WRSK. This was not by design but merely what the unit had in stock at the time of testing. As a result, no conclusions should be drawn about the choice of this particular set of WRSK spares. Another choice of 71% of the WRSK could result in a better or worse overall performance. Coronet Warrior was used to compare real world performance with the Dyna-METRIC predictions and not to assess this WRSK composition.

	NUMBER OF PARTS (AIRCRAFT ONLY)
WRSK AUTHORIZATIONS (100%)	3094
STARTING WRSK INVENTORY (71%)	2187

	PREDICTED	ACTUAL
DEMANDS	2162	946
ISSUES	1690	772
BALANCE (WITHOUT REPAIR)	497	1415

Figure 2. WRSK Demands
(Reprinted from 7)

The next figure gives a break out by day of the sorties tasked, the actual sorties flown, and the number of sorties predicted by the model.

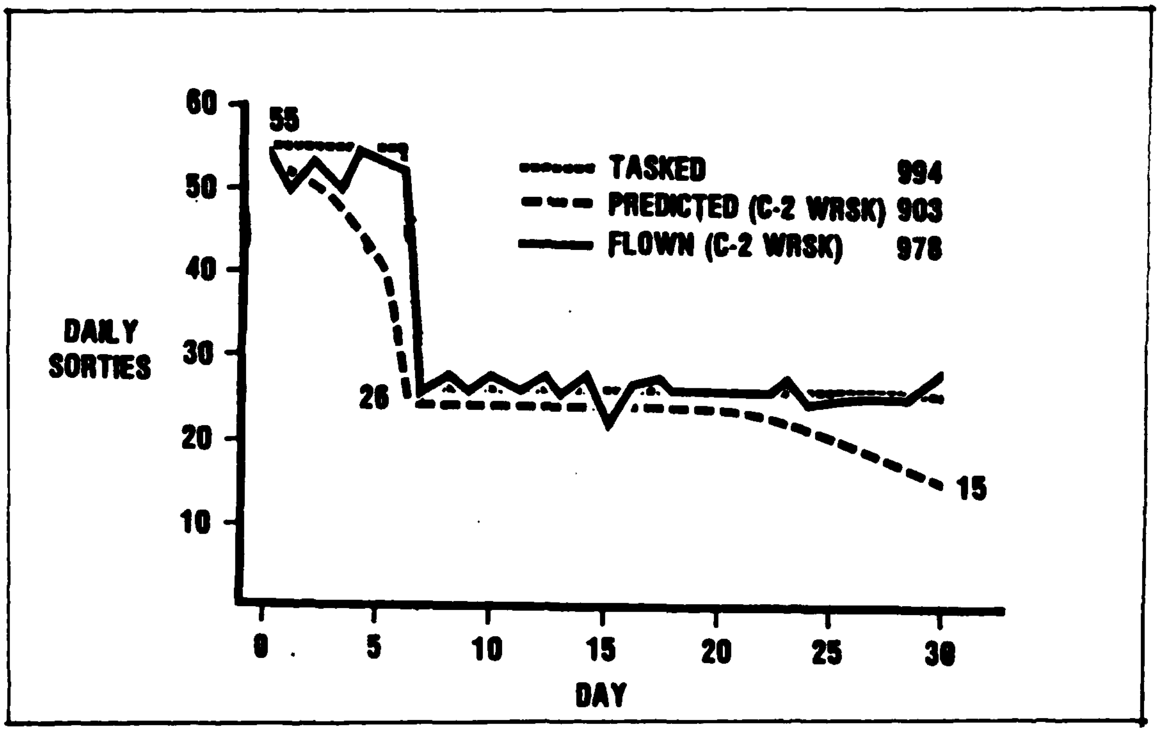


Figure 3. Sortie Performance
(Reprinted from 7)

Figure 4 shows the number of fully mission-capable aircraft by day both actual and predicted against the TAC fully mission-capable goal with 100% of the WRSK.

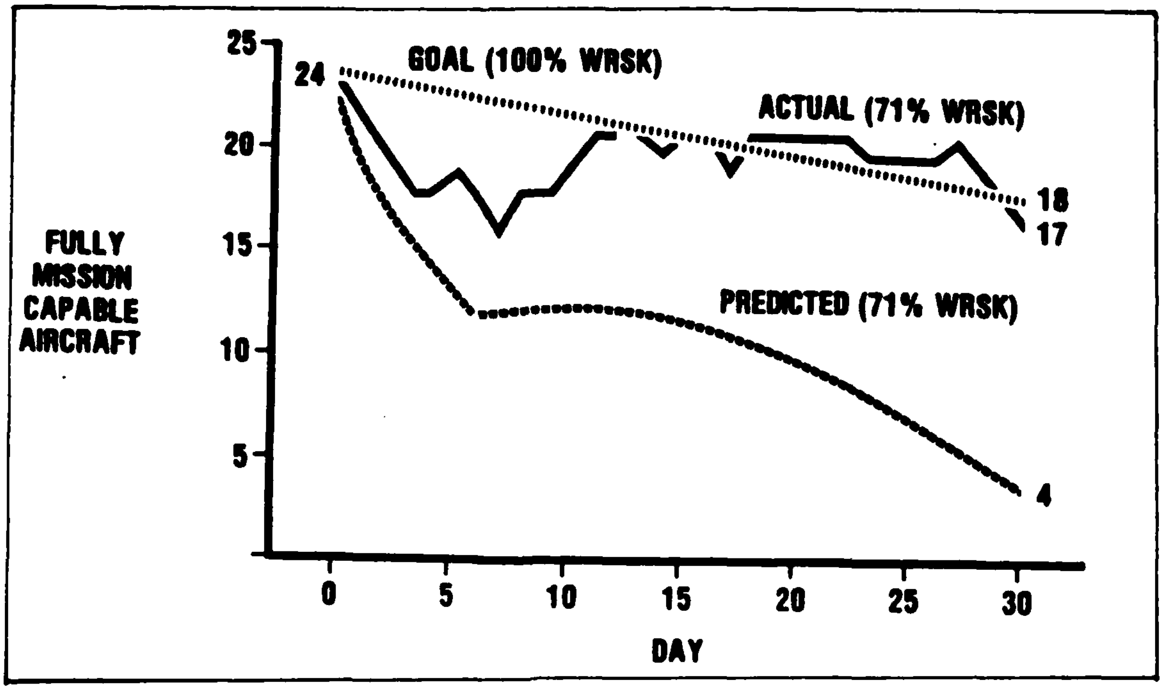


Figure 4. FMC Aircraft Performance
(Reprinted from 7)

Innovation was a key driver in the results shown. For example, repair capability was better than predicted. The exercise also affirmed some optimistic model assumptions such as those involving the avionics intermediate repair shop productivity and cannibalization effectiveness (7). A number of conclusions from this study are important to this work. First, the study concluded that the basic mechanics of Dyna-METRIC were sound. The demand and repair databases, however, need attention. Also, the ease of maintenance and cannibalization repair times should have more emphasis when setting stock levels. The study also concluded that a better and probably cheaper F-15 WRSK could be built. Finally, a similar exercise in 1988 was recommended for the

F-16 and continued use of Dyna-METRIC for sparing requirements was affirmed (7).

3. METHODOLOGY

The overall approach can be broken into several areas to include data gathering, model development and validation, and a data analysis plan. Basically, the research will identify shortfalls of the TAC Dyna-METRIC model and then find a software package that will emulate Dyna-METRIC. In addition, the model will also be capable of addressing some of the present limitations of Dyna-METRIC. The approach in each area will now be covered.

Data gathering is required in a number of areas. HQ TAC/LGY will provide their expertise in assessing the capabilities and shortfalls of the present model. HQ TAC/LGY will provide other data to include the input format and actual data from the Dyna-METRIC model providing the baseline input for the thesis. They will also provide requirements for the model output format. In searching for a suitable microcomputer language, current literature will be reviewed. As a part of this effort, HQ TAC/LGY will provide guidelines for hardware and software needs. Using the standard input data base mentioned earlier, HQ TAC/LGY will provide computer runs of Dyna-METRIC. The output from these runs will be necessary for comparison with the modeling effort. Also, HQ TAC/LGY will provide the results of the Coronet Warrior exercise to include maintenance data, exercise performance data, and output from a Dyna-METRIC run. The maintenance data will include demand rates, not repairable this station rates, quantity per application figures, repair cycle times and levels of indenture. A list of five parts with their individual repair cycle times should be included. The exercise data and Dyna-METRIC output should include the number of sorties and the number of fully mission-capable aircraft. Any deviations from the scenario or from the Dyna-METRIC model assumptions should be mentioned. This data gathering will provide information for the model development and data analysis portions of the study.

Model development and validation consists of a number of steps. First, the problem must be clearly defined. This should be done with a thorough literature review and HQ TAC/LGY user input. With this information and an appropriate software package, a model flow to emulate Dyna-METRIC will be developed. This outline will be written with the technical advice and approval of TAC. When the model flow is completed and approved, it will be coded and executed.

The model will be broken into three areas: the scheduling module, the sortie generation module, and the repair module. The scheduling module will be constructed as follows. Each day a daily flight schedule will be generated consisting of the sortie generation rate and sortie duration time. The number of sorties flown that day in the model is set to zero. The number of fully mission-capable aircraft is counted and stored. Then the daily sortie requirement is calculated. In Dyna-METRIC, daily sortie requirements are calculated by multipling the number of aircraft initially deployed by the required daily sortie generation rate. These requirements are not affected by the number of aircraft that might be available on that day. Next, the aircraft hangar is opened and aircraft are allowed to begin flying sorties. At the end of the day the aircraft hangar is closed and daily flying statistics are gathered. The statistics include the number of sorties flown and the number of fully mission-capable aircraft available. Although Dyna-METRIC only counts the number of fully mission-capable aircraft at the end of each day, because of the dynamic nature of the simulation model, this thesis calculates a daily average number of fully mission-capable aircraft (FMCA) based on (1).

$$FMCA = \frac{\underset{\text{beginning of day}}{\text{\# FMC aircraft}} + \underset{\text{end of day}}{\text{\# FMC aircraft}}}{2} \qquad (1)$$

This basic flow repeats itself for each day of the 30 day scenario.

The sortie generation module will be built as follows. At the beginning of the simulation, the number of deployed aircraft will be generated. As part of this generation each aircraft will be given a tail number. Also, each aircraft will be given a full complement of line replaceable units (LRU) and placed in a fully mission-capable status. This assumption is also made at the beginning of the Dyna-METRIC model. The aircraft are then sent to fly the first day's sorties. Before each sortie is flown a number of conditions must be met. First, the aircraft must be capable of finishing the sortie before the end of the day. Although this condition is also true in Dyna-METRIC, there is a slight difference because this thesis only allows complete sorties where Dyna-METRIC, because it is deterministic, aggregates partial sorties into complete ones. Second, if the daily sortie goal is met, the aircraft is returned to the hangar. This is identical to the Dyna-METRIC model which doesn't allow sorties to be flown past the daily sortie goal. Finally, if the sortie goal has not been met, then the average number of sorties flown by each aircraft for that day is checked. The aircraft cannot average more sorties per day than the maximum sortie generation rate of 3.5. If this average is exceeded, the aircraft is sent to the hangar. Dyna-METRIC has the same restriction but again aggregates this over all aircraft versus doing it by individual aircraft. If the aircraft is not sent back to the hangar it is ready to fly a sortie. The individual aircraft sortie rate is incremented and so is the daily sortie count. Then the aircraft begins the sortie. When the aircraft finishes the sortie, it calculates the number of demands per sortie. Based on this demand rate, a decision is made as to whether the aircraft is broken or not. If the aircraft is broken, it is sent to the repair module. If it is not broken, the aircraft is sent back to the flight line, available to fly again. When the end of the day is reached or all required sorties have been flown, the aircraft are sent to wait at the hangar until the next day's flying begins. At the beginning of the next day, the individual aircraft sortie rates are cleared and the aircraft are sent to fly again.

The repair module is constructed as follows. When aircraft come from the flight line, the time between failure is collected and the time of failure is noted. The aircraft then waits for a maintenance man and test stand. In the emulation mode this is an infinite resource to match the assumption made in Dyna-METRIC. After receiving a maintenance man, the aircraft enters the repair cycle. To emulate Dyna-METRICS, the repair cycle is instantaneous unless the aircraft must wait for a LRU. This part of the model will be flexible to allow insertion of realistic repair times for such things as remove and replace times, trouble shooting times, and transportation times. This is in addition to the parts waiting times that are already included in the model. The actual repair cycle begins by calculating what portion of the breaks are due to the selected LRUs and what portion are due to the rest of the aircraft. When this is done, a series of random draws are made and based on these draws the number and locations of the breaks are determined. If none of the selected LRUs are chosen, the aircraft is sent to the generic aircraft repair cycle. In this generic cycle, some proportion of the aircraft wait an average repair cycle time and then are sent to the flight line. This proportional split is set by the user based on experience. Those that don't wait are sent back to the flight line immediately. Before returning to the flight line each aircraft releases its maintenance resources. Those aircraft that have failures in both the selected LRUs and the generic aircraft LRUs proceed as if they only had broken selected LRUs because the broken selected LRUs are considered the critical assets. The aircraft with broken selected LRUs are sent to the selected LRU repair cycle. In the selected LRU repair cycle, the broken LRU is removed and sent to the intermediate level repair cycle. Then, based on whether cannibalization is allowed or not, the path splits. If cannibalization is not allowed, the maintenance resources are released and the aircraft waits for parts. When the parts become available, the aircraft gets a maintenance asset and is repaired. Then the aircraft releases the maintenance assets, reports time in repair, and returns to the flight line. If cannibalization is allowed, the broken LRU stock level is checked to see if parts are available. If they are, the aircraft is repaired and sent back to the flight line after releasing the maintenance assets and collecting time in repair. If the broken LRU is not available, the aircraft is disassembled and its parts are added to the available stock levels. The maintenance assets are also released for other uses. When parts become available, either from another aircraft or the intermediate level repair cycle, the aircraft waits for maintenance assets, is assembled, and returned to the flight line after releasing the maintenance assets and noting the time in the repair cycle.

The LRUs sent to the intermediate level repair cycle proceed as follows. Arriving LRUs are split by their not repairable this station (NRTS) rates. Those not repairable are not released back to the available stock. Those LRUs that are repairable are separated based on whether they have selected SRUs or not. If they don't have any selected SRUs they are sent on to be repaired. If they have selected SRUs, then a random draw is made and the broken SRU is chosen. If no SRU is chosen or the SRU is available in stock, the LRU is sent to be repaired. If no SRU is available, the LRU is disassembled and its SRUs are made available to other LRUs. When the LRUs are sent to be repaired, their repair time is based on a random draw from an exponential distribution using the repair cycle time as the mean parameter. After completing the repair, the LRU is sent back to the supply stock. No SRU repair capability exists at the unit level.

The following is a description of how the model chooses a broken selected LRU and SRU. When the selected LRUs or SRUs are read from the input file, the individual demand rate is multiplied by the quantity per aircraft for each part and these values are summed. Each part's proportion of the total demand rate is also calculated. This sets up a continuous distribution of demand fractions between zero and one. Then, when a part fails, a random draw between zero and one is made. This draw is compared to the zero-one distribution of demand rates. The interval where that draw falls determines the broken part. This method allows multiple duplicate failures and parallel processing of broken LRUs or SRUs.

The Dyna-METRIC model uses an exponential distribution for repair times. Questions have been raised as to whether this is a reasonable assumption. The ASAM model mentioned in Chapter Two, for example, uses a lognormal distribution for repair times. In order to study this assumption more closely, five LRUs with multiple repair times will be chosen for testing from the Coronet Warrior exercise. These repair times will form the input data files to be used with the AID package built by Pritsker and Associates. The repair data will be tested for goodness of fit with the lognormal and exponential distributions. Using both the chi-square and

Kolmogorov-Smirnov tests, runs will be made at the 95% significance level and the results will be reported. The data analysis plan flows out of the above model development and testing.

The data analysis plan revolves around the choice of software packages and the two groups of test runs. The choice of software packages will be based on user requirements and the availabilty of the software. User criteria will be rank-ordered. The package that is available and satisfies the most critical criteria will be used.

To use this model properly, a number of steps should be followed. The first step is to check the model assumptions and scenario to insure that these are appropriate for the present application. The following scenario applies to this model. It closely follows the Dyna-METRIC model scenario. The model calls for a 30 day deployment. Although the flight schedule can be changed, for the runs in this thesis, the first seven days have a requirement of 2.3 sorties per aircraft per day and a sortie duration of 1.8 hours. From day eight to day thirty, the number of sorties per aircraft per day is 1.1 and the sortie duration is still 1.8 hours. These flight requirements were provided by HQ TAC/LGY. Because of SLAM PC limitations, there are also limits on the number of LRUs and SRUs that can be modeled. The model allows the modeling of 40 separate LRUs. Those LRUs that are not explicitly modeled as a resource will be combined into one aircraft unit which will fail based on the combined demand rate of all the LRUs included in this aircraft unit. The portion of downtime associated with an aircraft unit failure will be the weighted average repair cycle time of all the LRUs included. The percentage of aircraft unit failures which actually cause model downtime will be set by the user based on experience. For this work the downtime percentage will be 2.5% for the TAC sample data and 1.25% for the Coronet Warrior data. The reason for the difference is that the TAC sample data has a much larger set of LRUs that are aggregated in the aircraft than the Coronet Warrior data does. LRUs that are modeled separately will use their own individual repair cycle times and will cause downtime only when a spare LRU is not available for replacement. The number of SRUs is limited to 30. Any SRU not being modeled is assumed to be available in sufficient quantity to meet anticipated demand (i.e. not a logistics driver). The model makes a number of assumptions in order to emulate the Dyna-METRIC model. The following assumptions are important. First, 100% cannibalization is not only assumed but all cannibalizations are considered successful and instantaneous. Every LRU is considered mission essential and equally important for maintenance. Maintenance is unconstrained. No resupply is allowed during the 30 day deployment and since this model is only at the squadron level, parts which are not repairable this station (NRTS) become unavailable for the rest of the deployment. Finally, the model doesn't allow attrition (20:19.25). Although Dyna-METRIC only looks at problem LRUs, this model will look at both LRUs and SRUs. The SLAM model being built will also be flexible enough that all of the above assumptions will be capable of being made more flexible. For example, the model has the capability to constrain maintenance and give individual LRU cannibalization/no cannibalization flags with cannibalization delays. The following assumptions are also made:

- The demand process is poisson.
- The demand process is independent of the repair process.
- The repair process uses times drawn from an exponential distribution with a given mean repair cycle time (20:4.6).

One assumption that is not in Dyna-METRIC but is in this model is that no more than two failures are allowed per sortie (i.e. the total aircraft demand rate per flying hour times the sortie duration cannot be larger than two). This is not considered restrictive since the demand data provided has been well below this restriction.

Validation and verification is very important. Verification is showing that the model behaves as intended (9:11). To show this, Capt. Moulder, another graduate student, will take the logic flow diagrams and list of variables and compare them to the written code. If any discrepancies are found, they will be noted and corrected. This process will continue iteratively until the model logic matches the written code. Model validation, which is building the user's confidence to an acceptable level, will involve a number of steps (28:33). These steps include checking the face value and assumptions of the model and testing the input-output transformations of the model against the sample input and output provided by TAC and the input-output transformation from the Coronet Warrior exercise. The face value testing will be an iterative process where the model flow is built, showed to the user, changes made, and then showed to the user again. When the user is satisfied that the model flow and assumptions match the Dyna-METRIC model in sufficient detail, the face value testing is complete. The testing of the input-output transformations will be more difficult. Each comparison test will be run in a similar way (ie. model vs. TAC data and model vs. Coronet Warrior data).

Two model outputs will be tested. The first will be the average sorties generated. The second will be the average number of fully-mission capable aircraft. Because there are two output parameters of interest, a technique to compare the two parameters simultaneously is needed. In addition, the output varies between days. In order to do this, the following method will be used. TAC has set the standard deviation of the test runs at one aircraft per day and 1.25 sorties per day. A pilot sample of five runs will be taken. Based on this sample the appropriate number of runs will be calculated to get the desired accuracy. The model will then be rerun to achieve the desired accuracy. The output from these runs will be averaged for each day of the deployment. These daily averages will be subtracted from the Dyna-METRIC expected-value daily averages. The differences will then form the database for a two-sided hypothesis test using the normal statistic. The null hypothesis will be that the difference in model means equals zero while the alternate hypothesis will be that the difference in means does not equal zero. Each test will use a 95% significance level. For the model to be accepted, the test should not reject the null hypothesis. Conclusions will be made as to why differences do or do not exist.

4. RESULTS

The first task mentioned in Chapter Three is to identify capabilities and limitations of the Dyna-METRIC model. Some of these capabilities and limitations have already been mentioned in Chapter One in reference to the problem background and research objectives. Additional capabilities and limitations were mentioned in Chapter Two in the Dyna-METRIC section. For example, one capability of the Dyna-METRIC model is its flexibility as a tool for analyzing the impact of logistics sparing on aircraft capability (15:1). Limitations such as unconstrained maintenance and unlimited Depot supply are also mentioned (15:1). Assumptions and limitations are summarized in Appendix C.

HQ TAC has provided a number of important data items. The first is an input data set for the F-15E which is being used at HQ TAC. Along with this input file, HQ TAC provided a sample Dyna-METRIC output using this input data. Only one run of the model is necessary because of the deterministic nature of Dyna-METRIC. In addition, they provided similar input and output files for the Coronet Warrior exercise. Due to the length and complexity of these data sets, only the most important portions have been included in this thesis. They can be found in Appendix D. The content of these data sets are explained in this Appendix. For copies of the complete input and output files contact HQ TAC/LGY.

The next task was to identify hardware and software needs, prioritize these needs, collect information on hardware and software availability and then choose an appropriate package for use. HQ TAC identified a number of system requirements. The hardware requirements include the capability to run on a Z-248 with 640K RAM, a single floppy disk, a single, 10MB hard disk and a wide carriage dot matrix printer. This should be done with minimum additional investment in hardware (15:3). The software package should be economical, capable of emulating Dyna-METRIC with execution times of less than 2 hours, and capable of looking at 50 to 100 problem items. It should also have graphics modification capability. Several restrictions were also placed on the choice because of the thesis requirements. For example, the software must be readily available to the author and either easily learned or already known by him because of the short time allotted for the thesis effort. In accessing the available software, the information in Chapter Two under Microcomputers provided the primary basis for comparison. Based on the above considerations, especially availability of software, cost, wide-spread usage and acceptance of the language and knowledge of the language by this author, HQ TAC chose and supplied SLAM II PC as the model development software.

In developing the model and testing it, HQ TAC identified two key output parameters of interest. They were the number of sorties flown per day and the number of fully mission-capable aircraft available (5).

Before discussing the specific results of the testing, the reader is again reminded of the assumptions and limitations of both Dyna-METRIC and this thesis model. These assumptions were discussed in Chapter Three. Also a summary is given in Appendix C for reference.

The Coronet Warrior exercise was discussed in Chapter Two. The important results were as follows:

1) Demand for parts was lower than predicted.
2) Parts were fixed faster and more reliably than predicted.
3) Personnel adapted and performed better than predicted.
4) Innovation by personnel was a key driver in the results.
5) Optimistic model assumptions were affirmed including:
    - avionics intermediate repair shop productivity.
    - cannibalization effectiveness.
6) Basic mechanics of Dyna-METRIC are sound.
7) Demand and repair data bases need attention (7).

More extensive data sets and results are available from HQ TAC. The Coronet Warrior input and Dyna-METRIC output files are a direct result of this exercise (see Appendix D).

HQ TAC provided five sets of repair times for testing repair time distributions. These times can be found in Appendix E. Two types of goodness-of-fit tests were performed. They were the Chi-square method and the Kolmogorov-Smirnov (K-S) method. Since the K-S method is generally accepted as more powerful only the results from this test were shown here (6:346). The full results were given in Appendix E. The null hypothesis for these tests was that they were either exponentially or lognormally distributed. The alpha level was set at 0.05 or a 95% confidence level. The K-S test results are shown in Table 3. They are broken out by data set and distribution type.

Table 3. Repair Time Distribution Tests

		K-S TEST STATISTIC	
DATA SET	CRITICAL VALUE	EXPONENTIAL	LOGNORMAL
1	0.287	0.209	0.119
2	0.234	0.129	0.096
3	0.275	0.173	0.218
4	0.203	0.151	0.095
5	0.194	0.157	0.068

NOTE: ALPHA=0.05

The verification effort was concluded by Capt. Moulder on November 16, 1987 when model flow diagrams and SLAM II coding were compared for discrepancies. No differences were found. The flow charts and the SLAM code can be found in Appendix B. A description of the basic model flow can be found in Chapter Three. The model code also has internal documentation.

Face-value validation took place through a number of work sessions and personal interviews with Capt. Budde, HQ TAC/LGY and Capt. Mabe, AFIT/LSMA (5) (21). Final agreement on the model flow took place at a meeting on 30 September 1987 with Capt. Budde at HQ TAC. This model flow was then implemented in SLAM II. The approved flow is the previously mentioned one in Appendix B.

Results from the testing of input-output transformations involved a number of steps. First, five pilot runs were made for the TAC sample and Coronet Warrior input data. Each run produces one data point for each day. Means and standard deviations for each day were calculated over the five runs. Tables 4 and 5 show the means and standard deviations by day and output category. Table 4 is for the TAC sample data and Table 5 is for the Coronet Warrior data.

Table 4. Output Variances For 5 Runs TAC Sample Data

	OUTPUT			
	SORTIE		FMC	
DAY	MEAN	STD. DEV.	MEAN	STD. DEV.
1	55.00	0.00	22.40	0.37
2	55.00	0.00	19.90	0.76
3	55.00	0.00	17.60	0.78
4	53.60	1.17	15.30	0.98
5	50.20	2.85	13.40	0.70
6	46.40	2.71	12.00	0.79
7	40.80	2.92	11.00	0.74
8	26.00	0.00	11.10	0.91
9	26.00	0.00	12.10	0.53
10	26.00	0.00	12.50	0.74
11	26.00	0.00	13.10	1.29
12	26.00	0.00	14.00	1.24
13	26.00	0.00	14.20	1.03
14	26.00	0.00	13.70	0.85
15	26.00	0.00	13.30	0.93
16	26.00	0.00	13.00	1.19
17	26.00	0.00	12.70	1.19
18	26.00	0.00	12.50	0.88
19	26.00	0.00	11.80	0.64
20	26.00	0.00	11.30	0.51
21	26.00	0.00	10.50	0.67
22	25.80	0.20	9.30	0.80
23	25.60	0.40	8.50	0.91
24	23.60	1.50	7.80	1.20
25	22.00	2.51	7.20	1.38
26	20.60	2.98	6.50	1.46
27	19.60	2.88	6.40	1.36
28	20.20	2.60	6.30	1.32
29	20.20	2.85	6.40	1.43
30	20.00	2.97	6.10	1.33

Table 5. Output Variances For 5 Runs Coronet Warrior Data

	OUTPUT			
	SORTIE		FMC	
DAY	MEAN	STD. DEV.	MEAN	STD. DEV.
1	55.00	0.00	22.90	0.19
2	55.00	0.00	21.00	0.35
3	55.00	0.00	20.10	0.51
4	55.00	0.00	20.20	0.41
5	55.00	0.00	19.80	0.68
6	55.00	0.00	19.20	0.72
7	55.00	0.00	18.80	0.58
8	26.00	0.00	18.40	0.29
9	26.00	0.00	18.60	0.29
10	26.00	0.00	19.00	0.57
11	26.00	0.00	19.30	0.80
12	26.00	0.00	19.50	0.80
13	26.00	0.00	19.60	0.83
14	26.00	0.00	19.50	0.79
15	26.00	0.00	19.40	0.68
16	26.00	0.00	19.30	0.66
17	26.00	0.00	19.00	0.76
18	26.00	0.00	18.50	0.82
19	26.00	0.00	18.40	0.83
20	26.00	0.00	18.50	0.84
21	26.00	0.00	18.40	0.66
22	26.00	0.00	18.30	0.58
23	26.00	0.00	18.00	0.76
24	26.00	0.00	17.70	0.86
25	26.00	0.00	17.30	0.93
26	26.00	0.00	16.60	1.01
27	26.00	0.00	16.30	1.04
28	26.00	0.00	16.50	1.06
29	26.00	0.00	16.50	1.02
30	26.00	0.00	16.20	1.01

HQ TAC requirements called for a standard deviation for sorties of less than 1.25 and a standard deviation for fully mission capable aircraft of less than 1.0 (5). Since neither pilot run met the requirement, 15 runs were made. Sample results are shown in Table 6 and Table 7.

Table 6. Output Variances For 15 Runs TAC Sample Data

	OUTPUT			
	SORTIE		FMC	
DAY	MEAN	STD. DEV.	MEAN	STD. DEV.
1	55.00	0.00	22.30	0.15
2	55.00	0.00	19.40	0.38
3	54.50	0.40	17.00	0.54
4	52.70	0.95	15.10	0.60
5	49.50	1.49	13.90	0.51
6	46.90	1.59	12.90	0.49
7	44.80	1.77	12.00	0.48
8	26.00	0.00	11.80	0.48
9	26.00	0.00	12.40	0.51
10	26.00	0.00	12.50	0.63
11	26.00	0.00	12.70	0.74
12	26.00	0.00	13.50	0.65
13	26.00	0.00	13.70	0.62
14	26.00	0.00	13.40	0.65
15	25.90	0.00	13.00	0.64
16	25.80	0.20	12.60	0.70
17	25.80	0.20	12.60	0.66
18	26.00	0.00	12.50	0.53
19	26.00	0.00	12.00	0.55
20	26.00	0.00	11.50	0.54
21	26.00	0.00	10.80	0.55
22	25.90	0.07	9.90	0.61
23	25.70	0.18	9.40	0.64
24	24.70	0.64	9.00	0.72
25	24.30	0.92	8.60	0.79
26	22.90	1.46	8.10	0.84
27	22.50	1.57	7.80	0.77
28	22.50	1.70	7.60	0.75
29	22.00	1.65	7.40	0.79
30	21.50	1.72	6.90	0.77

Table 7. Output Variances For 15 Runs Coronet Warrior Data

	OUTPUT			
	SORTIE		FMC	
DAY	MEAN	STD. DEV.	MEAN	STD. DEV.
1	55.00	0.00	23.10	0.13
2	55.00	0.00	21.70	0.23
3	55.00	0.00	21.00	0.28
4	55.00	0.00	20.90	0.23
5	55.00	0.00	20.50	0.27
6	55.00	0.00	20.10	0.33
7	55.00	0.00	19.60	0.30
8	26.00	0.00	19.50	0.33
9	26.00	0.00	19.70	0.34
10	26.00	0.00	19.90	0.32
11	26.00	0.00	19.80	0.32
12	26.00	0.00	19.80	0.26
13	26.00	0.00	19.90	0.31
14	26.00	0.00	19.70	0.33
15	26.00	0.00	19.50	0.34
16	26.00	0.00	19.50	0.30
17	26.00	0.00	19.30	0.32
18	26.00	0.00	18.90	0.34
19	26.00	0.00	18.60	0.34
20	26.00	0.00	18.40	0.34
21	26.00	0.00	18.40	0.27
22	26.00	0.00	18.40	0.28
23	26.00	0.00	18.20	0.28
24	26.00	0.00	18.20	0.34
25	26.00	0.00	17.70	0.34
26	26.00	0.00	17.40	0.38
27	26.00	0.00	17.40	0.41
28	26.00	0.00	17.10	0.40
29	26.00	0.00	17.00	0.40
30	26.00	0.00	16.70	0.40

The 15 runs still did not reduce the standard deviations to the desired levels. In response to this, a test using 50 runs was completed. Results are displayed in Tables 8 and 9.

Table 8. Output Variances For 50 Runs TAC Sample Data

	OUTPUT			
	SORTIE		FMC	
DAY	MEAN	STD. DEV.	MEAN	STD. DEV.
1	55.00	0.00	22.60	0.08
2	55.00	0.00	19.90	0.21
3	54.80	0.13	17.80	0.29
4	53.40	0.43	16.00	0.32
5	51.10	0.69	14.40	0.29
6	48.80	0.89	13.40	0.30
7	47.30	1.01	12.80	0.30
8	26.00	0.00	12.50	0.30
9	26.00	0.00	13.10	0.28
10	26.00	0.00	13.50	0.31
11	26.00	0.00	13.50	0.35
12	26.00	0.00	13.70	0.34
13	26.00	0.00	13.70	0.33
14	26.00	0.00	13.40	0.35
15	26.00	0.02	13.10	0.33
16	25.90	0.06	12.90	0.32
17	25.90	0.06	12.80	0.31
18	26.00	0.00	12.40	0.31
19	26.00	0.00	12.00	0.32
20	25.90	0.12	11.50	0.30
21	26.00	0.04	10.90	0.30
22	26.00	0.02	10.40	0.31
23	25.80	0.09	10.00	0.32
24	25.30	0.30	9.50	0.34
25	24.60	0.39	8.90	0.36
26	23.60	0.60	8.20	0.38
27	22.90	0.68	7.60	0.37
28	22.10	0.74	7.20	0.36
29	21.40	0.75	7.00	0.37
30	21.30	0.81	6.70	0.36

Table 9. Output Variances For 50 Runs Coronet Warrior Data

	OUTPUT			
	SORTIE		FMC	
DAY	MEAN	STD. DEV.	MEAN	STD. DEV.
1	55.00	0.00	23.10	0.06
2	55.00	0.00	21.80	0.13
3	55.00	0.00	21.10	0.14
4	55.00	0.00	20.60	0.16
5	55.00	0.00	20.10	0.18
6	55.00	0.00	19.60	0.21
7	54.90	0.06	19.40	0.20
8	26.00	0.00	19.40	0.21
9	26.00	0.00	19.60	0.19
10	26.00	0.00	19.70	0.18
11	26.00	0.00	19.80	0.17
12	26.00	0.00	19.70	0.15
13	26.00	0.00	19.60	0.17
14	26.00	0.00	19.50	0.18
15	26.00	0.00	19.40	0.19
16	26.00	0.00	19.20	0.19
17	26.00	0.00	19.10	0.20
18	26.00	0.00	18.80	0.20
19	26.00	0.00	18.60	0.22
20	26.00	0.00	18.40	0.23
21	26.00	0.00	18.30	0.22
22	26.00	0.00	18.20	0.22
23	26.00	0.00	18.00	0.38
24	26.00	0.00	17.90	0.23
25	26.00	0.00	17.70	0.21
26	26.00	0.00	17.40	0.21
27	26.00	0.00	17.30	0.20
28	26.00	0.00	17.20	0.21
29	26.00	0.00	16.90	0.23
30	26.00	0.00	16.70	0.25

Plots of each run by day versus the number of sorties and versus the number of FMC aircraft are shown in Appendix F. These plots should give some idea of the variability of the research model for the 50 runs. These runs met the TAC variance requirements.

Using the data collected from the 50 runs, a comparison between the Dyna-METRIC model and the research model was performed. This was done by calculating the differences between output means day by day. These 60 differences were then averaged and a standard deviation calculated. Based on the Central Limit Theorem and the large sample size (38:279-280), a hypothesis test using the normal approximation was used (38:388-392). The following gives the test and test results. The Alpha level in both tests was equal to 0.05. The first test was from the TAC sample data. The second was from the Coronet Warrior data.

TEST 1: Ho: Dyna-METRIC model output - Research model output = 0.

Ha: Difference does not equal zero.

Test statistic:

$$Z = SD-HD/S$$

$$Z = 1.540 - 0/ 1.827 = 0.8429$$

Rejection Region:

$$|Z| > Z(\alpha/2)$$

$$|Z| > 1.96$$

where

SD = sample difference in model outputs

HD = null hypothesis difference in output means

S = sample standard deviation

These values were calculated using a fortran program and a SAS program given in Appendix G. The Z value was not in the rejection region and therefore, no significant difference exists between the research model output and the Dyna-METRIC model output.

TEST 2: Ho: Dyna-METRIC model output - Research model output = 0.

Ha: Difference does not equal zero.

Test statistics:

$$Z = SD - HD/\ S$$

$$Z = 0.0285 - 0\ /\ 0.440 = 0.0648$$

Rejection Region:

$$|Z| > Z(\ /2)$$

$$|Z| > 1.96$$

where

SD = sample difference in model outputs

HD = null hypothesis difference in output means

S = sample standard deviation

These values were calculated using a Fortran program and a SAS program given in Appendix G. The Z value was not in the rejection region and therefore, no significant difference exists between the research model output and the Dyna-METRIC model output.

The following two figures show the output differences between the research model and the Dyna-METRIC model for each output by deployment day. Figure 5 shows the differences in the number of sorties by day between the Dyna-METRIC model and the research model. The Ts are the differences betweeen the models for the TAC sample data and the Cs are the differences between the models for the Coronet Warrior data. Figure 6 shows the differences in the number of fully mission-capable aircraft by day between the Dyna-METRIC model and the research model. Again, the Ts represent the TAC sample data and the Cs represent the Coronet Warrior data. The actual calculated differences are given in Appendix G. The daily output data for the Dyna-METRIC model and the research model and plots of the appropriate calculated daily averages versus sorties and versus FMC aircraft for the research model are also given in Appendix G.

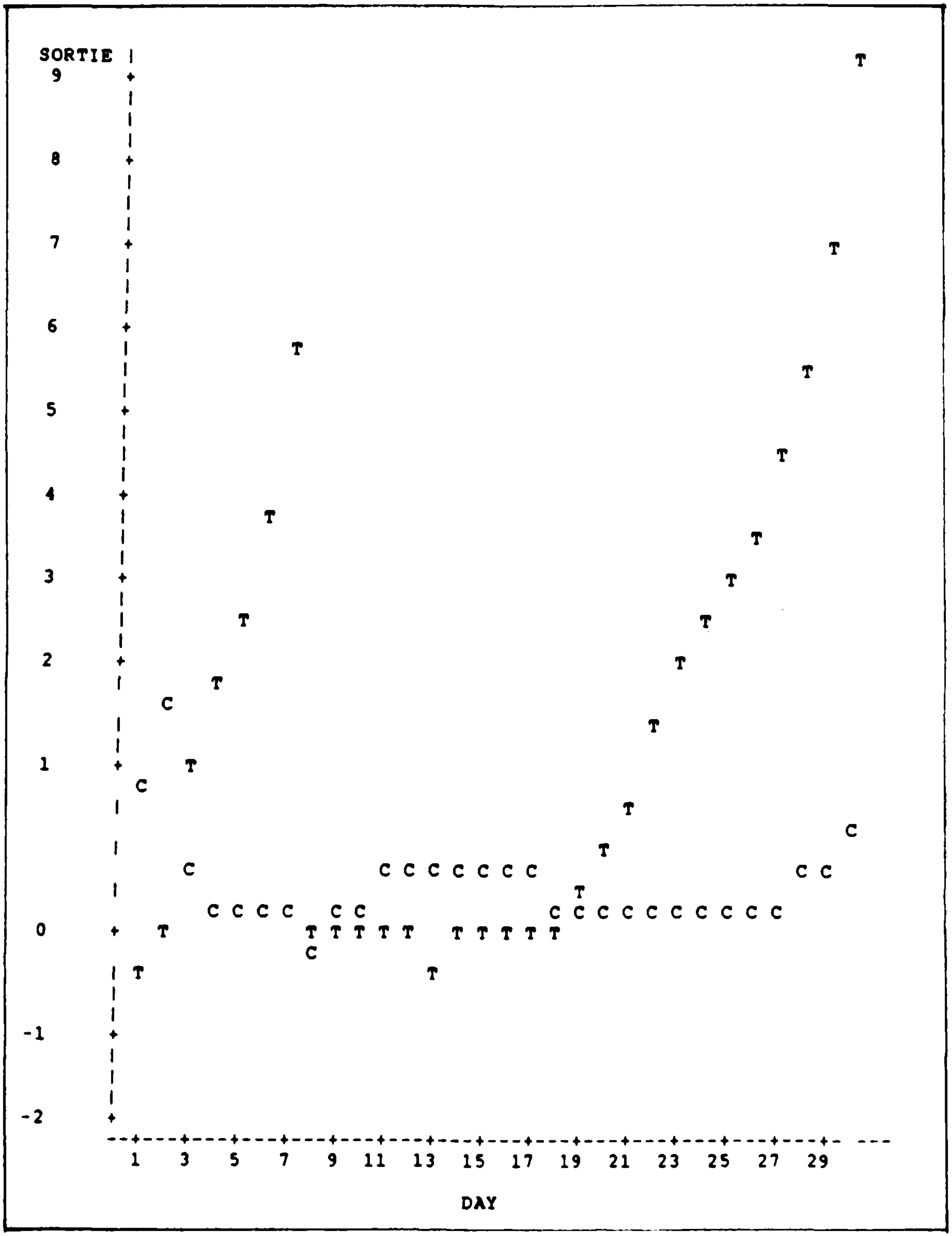


Figure 5. Sortie Output Differences

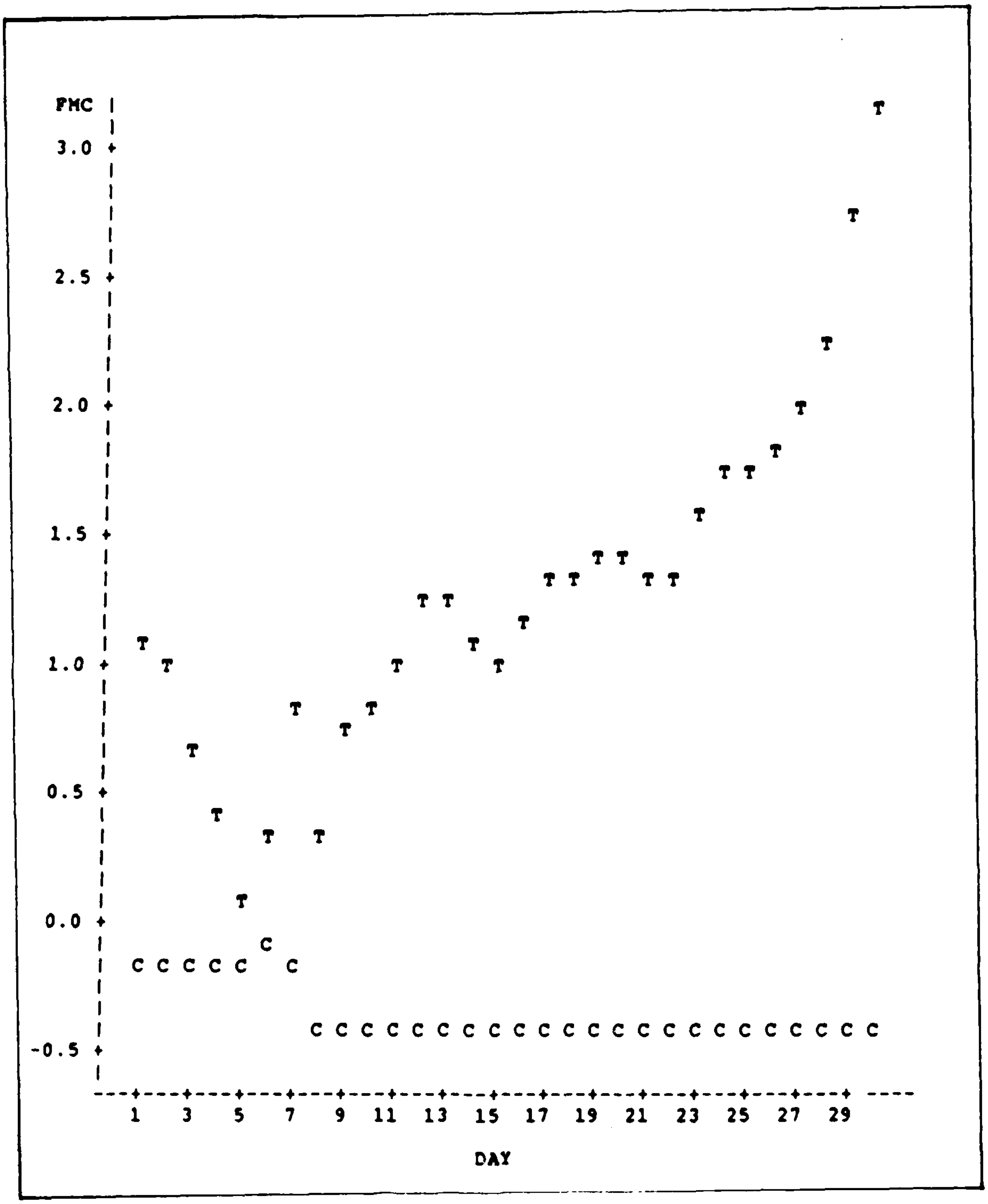


Figure 6. FMC Aircraft Output Differences

The results in Chapter Four present a clear picture of the research performed in this thesis effort. Chapter Five will draw some conclusions from the results and recommend future areas for research.

5. Conclusions and Recommendations

This chapter draws several conclusions based on the results given in Chapter Four and then presents some recommendations for future research.

Conclusions

First, in selecting SLAM II PC as the programming language, HQ TAC considered price, portability, widespread acceptance of the language, and knowledge by the author as primary reasons for using SLAM II PC. They also liked SLAM's optional graphics edit capability and the optional graphics simulation display. There are some short-comings. First, simulation on the PC is slow. Fifty runs takes almost an hour, averaging about a minute a run. Also, compiling and executing the PC version and then using the output is more difficult than with the mainframe version.

The results from the repair time distribution tests are inconclusive. Either distribution seems to fit the data in most cases. To make a distinction more repair time data should be collected.

The research model emulates Dyna-METRIC fairly well. The P-values for the two hypothesis tests are small. Looking at Figures 5 and 6 in Chapter 4, one thing should be noted. The only significant differences in results can be seen between day 5 and day 7 and after day 22 in the TAC sample data. The research model predicts higher results than Dyna-METRIC. This is encouraging since Dyna-METRIC output tends to be pessimistic in the later stages of the model run. This is because Dyna-METRIC bases failures on required sorties rather than actual sorties flown, thus more aircraft break and fewer sorties are flown than would be expected. Also, note that this dramatic decrease in performance was not evident in the Coronet Warrior exercise (7). A larger variance in the TAC sample results can also be attributed to the larger number of LRUs aggregated in the research model. Additional plots of the output data are available in Appendix G.

In addition to providing emulation capability on the PC, this model provides TAC with the capability to constrain maintenance personnel and equipment. It also gives the user the ability to break out maintenance times by individual tasks and provides enhanced scheduling capabilities. With the addition of SLAM upgrades, graphics modification and graphics display capabilities will be possible. The model does have some limitations. These limitations include limiting the number of LRUs and SRUs modeled and limiting the number of failures per sortie to two (see Appendix C).

In conclusion the completed research model provides HQ TAC with an analytical tool not available before on the PC.

It also provides maintenance constraints and scheduling flexibility that didn't exist in previous models. In addition, its initial emulation and favorable comparison with Dyna-METRIC provides easier acceptance of the model by TAC users.

Recommendations

There exist several areas of research which can grow from this basic work. First, more data can be collected and a study done on repair time distributions. Research should not only study the actual distribution of the repair times but the impact or significance of using one distribution versus another on the results of this model. Another area which could improve the results of this model is research into implementing variance reduction techniques in the model. This work could reduce the number of runs to achieve a specified accuracy. Another area of research is enhancing the processing of the model input and output. It could include direct user input into the model and customized report builders. The largest area for potential work is in sensitivity analysis using the enhanced capabilities of this model. The future researcher could implement, test, and draw conclusions on the impact of various maintenance and scheduling constraints on TAC fully mission-capable status and sortie generation capability. Sensitivity analysis on these results could have a significant effect on WRSK sparing levels, maintenance manning levels, and types and quantities of other equipment deployed with a unit.

Appendix A: Microcomputer Simulation Software

Table 14. Microcomputer Simulation Software (3:169-172)

SOFTWARE NAME	BRIEF DESCRIPTION	COMPUTERS	OPERATING SYSTEM	OTHER SOFTWARE	RAM	OTHER HARDWARE	MODELING CAPABILITIES	APPROXIMATE PRICE
ACES (All-purpose Continuous Equation Sim.)	Solves & displays algebraic and differential/integral equations. Uses a 4th order Runge-Kutta integration algorithm	Apple II			48K	Disk II Required	Continuous	$200
ACSL (Advanced Continuous Simulation Language)	Models the dynamic response of physical systems. Uses Gear's integration algorithm for stiff systems. Full compatibility with FORTRAN subroutines	IBM PC and compatibles	MS DOS 2.x or 3.x	Microsoft FORTRAN 3.2	256K	IBM, Hercules or equiv. graphics adaptor	Continuous	$1600/$1000 (quantity discount avail.)
EZQ	Solves differential, difference & algebraic equations, with graphical and tabular output	Apple II	DOS 3.3		64K	Printer optional	Continuous	$80
GASS (General Application Simulation System)	Simulates up to 10 vars. simultaneously and combines them into one user-defined algorithm. Random variables can be from 13 different probability distributions	Apple II, II+, IIe or III (emulation) IBM PC	DOS 3.3 DOS 2.0		48K 128K	Printer required		$325
GPSS/H	Full implementation of the state-of-the-art GPSS version. Features interactive debugging, flexibility, and high-speed execution	IBMPCXT/370 IBMPCAT/370 Micro VAX MC68000 IBM PC AT	CMS VMS & UNIX Xenix			Terminal compat. w/VT-100 (adv. video) rec. 20mb hard disk rec.	Discrete	$3500 per year/ $250

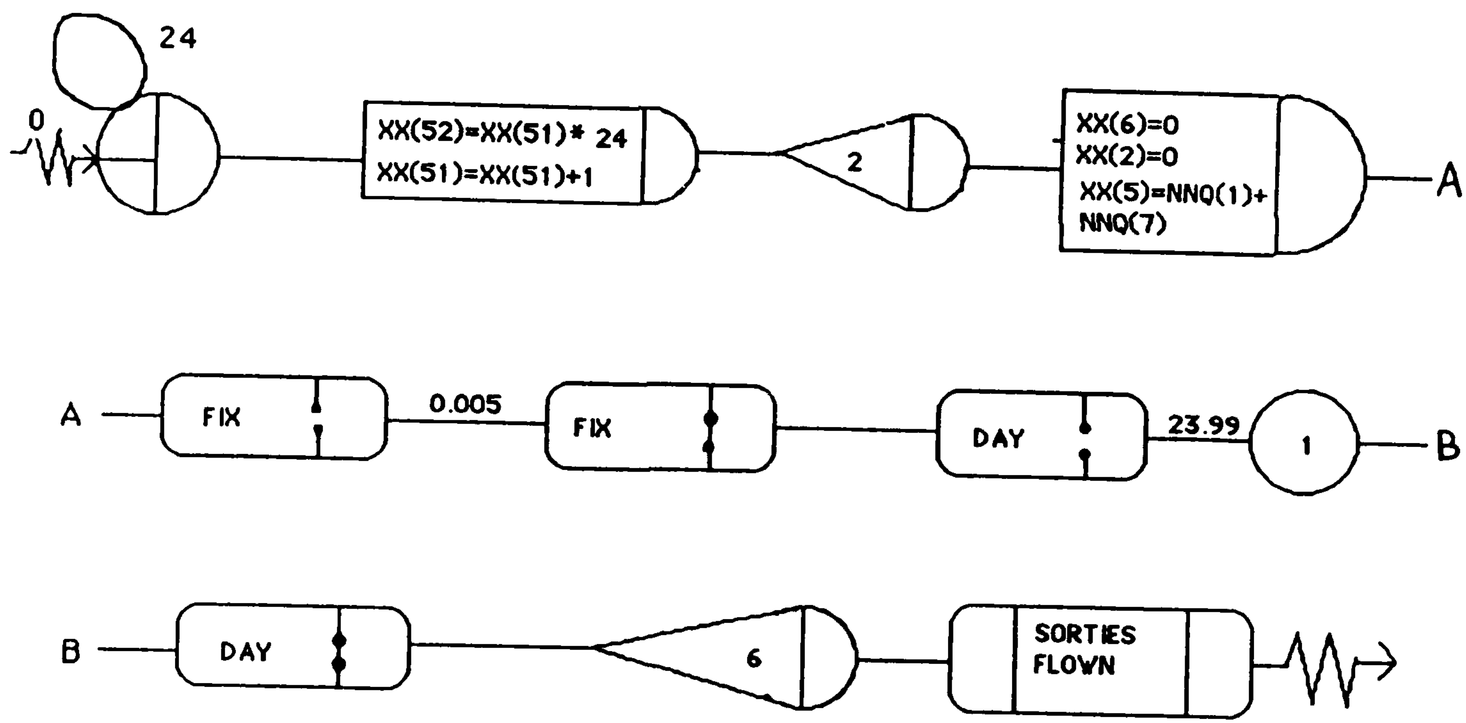
SOFTWARE NAME	BRIEF DESCRIPTION	COMPUTERS	OPERATING SYSTEM	OTHER SOFTWARE	RAM	OTHER HARDWARE	MODELING CAPABILITIES	APPROXIMATE PRICE
GPSS/PC	An interactive version of GPSS. Features include keystroke error prevention, interactive debugging, and mid-simulation modification	IBM PC IBM PC/XT	DOS 1.1, 2.0		256K		Discrete	\$900/\$900 (special discounts avail. to educ. inst.)
Interactive	Process-oriented models. For eval. of manufacturing systems, inventory policies, etc.	IBM PC IBM PC/XT Apple III	DOS 2.x DOS 3.3	UCSD Pascal	256K		Discrete	\$750/\$350
Inter-SIM	Discrete event visual interactive simulation package. Comprehensive file menu and visual interaction ease model development	Apple II IBM PC DEC Rainbow ACT Apricot	MS-DOS	UCSD Pascal			Discrete	235-315 pounds (mult. site licenses)
ISG (Interactive Scientific Graphics)	Package of UCSD Pascal procedures with interactive features. Plots sets of functions with full labeling	Apple II, II+		Pascal 1.1	64K	Lang. card or other 16K RAM card	Continuous	\$95
ISL (Interactive Simulation Languages)	Solves nonlinear differential equations. User can view graphic results and change parameter values during execution	Any Apple II	Apple DOS 3.3		48K		Continuous	\$500
Micro-DYNAMO	Compiles and simulates complex models of cause and effect relationships over time	Apple II IBM PC & compatibles		Pascal 1.1 none	64K 128K	Printer rec. Color mon. rec.; 2 disks req. for Apple, rec. for IBM	Continuous	\$245, Apple II; \$395 IBM PC
Micro-PASSIM	Pascal-based sim. modeling language. Can perform discrete, continuous and com-	Apple II, III IBM PC	MS DOS 2.x	Pascal 1.1 or UCSDp-sys. Vers. 2.x of Turbo	48K 128K		Discrete, continuous & combined	\$125

SOFTWARE NAME	BRIEF DESCRIPTION	COMPUTERS	OPERATING SYSTEM	OTHER SOFTWARE	RAM	OTHER HARDWARE	MODELING CAPABILITIES	APPROXIMATE PRICE
	bined simulation. Uses event scheduling and process orientation world views			Pascal				
Monte Carlo Simulations	Analyzes past results and forecasts the outcome of similar or modified undertakings. 7 prob. distributions are built in, along with the Chi-Square goodness of fit test	Apple II, II+, IIe or III (emulation) IBM PC	DOS 3.3 DOS 2.0		48K 128K	Printer rec.		$125
PC Model	A simulation system specifically designed to model the movement of manufactured assemblies through the assembly process	IBM (PC, XT, PCjr, portable, and AT) or compatible	PC DOS	Full screen editor (IBM's Personal or Prof. Editor) highly rec.	128K		Discrete	$450/$360 (quantity discounts avail.)
Scheduling Simulator	Designed for simulating job shops in order to analyze trial schedules	Apple Lisa & Macintosh	DOS 3.3			External Discrete disk dr. & printer req. Hard disk opt.	Discrete	$535
See Why	A visual, discrete event, interactive system. Consists of 180 high-level FORTRAN subroutines.	IBM PC/AT Cromenco	DOS 3.x CROMIX	IBM Prof. FORTRAN	640K		Discrete	$31,700
SIMAN	General purpose language. Includes special features for manufacturing systems. Offers three modeling orientations	IBM PC & compatibles	MS DOS 2.x	Microsoft FORTRAN 3.2 (not req. to run block models)	320K	2 disk drives	Discrete, continuous & combined	$1500/$200 (quantity discounts & mult. site licenses avail.)
SIMSCRIPT II.5	A full implementation of the mainframe version of SIMSCRIPT. Features include a built-in text editor and the use of virtual memory	IBM PC IBM PC/XT IBM PC/AT	MS DOS 2.x		320K	8087 5Mb hard disk	Discrete	$24,500/$250 (quantity discounts avail.)

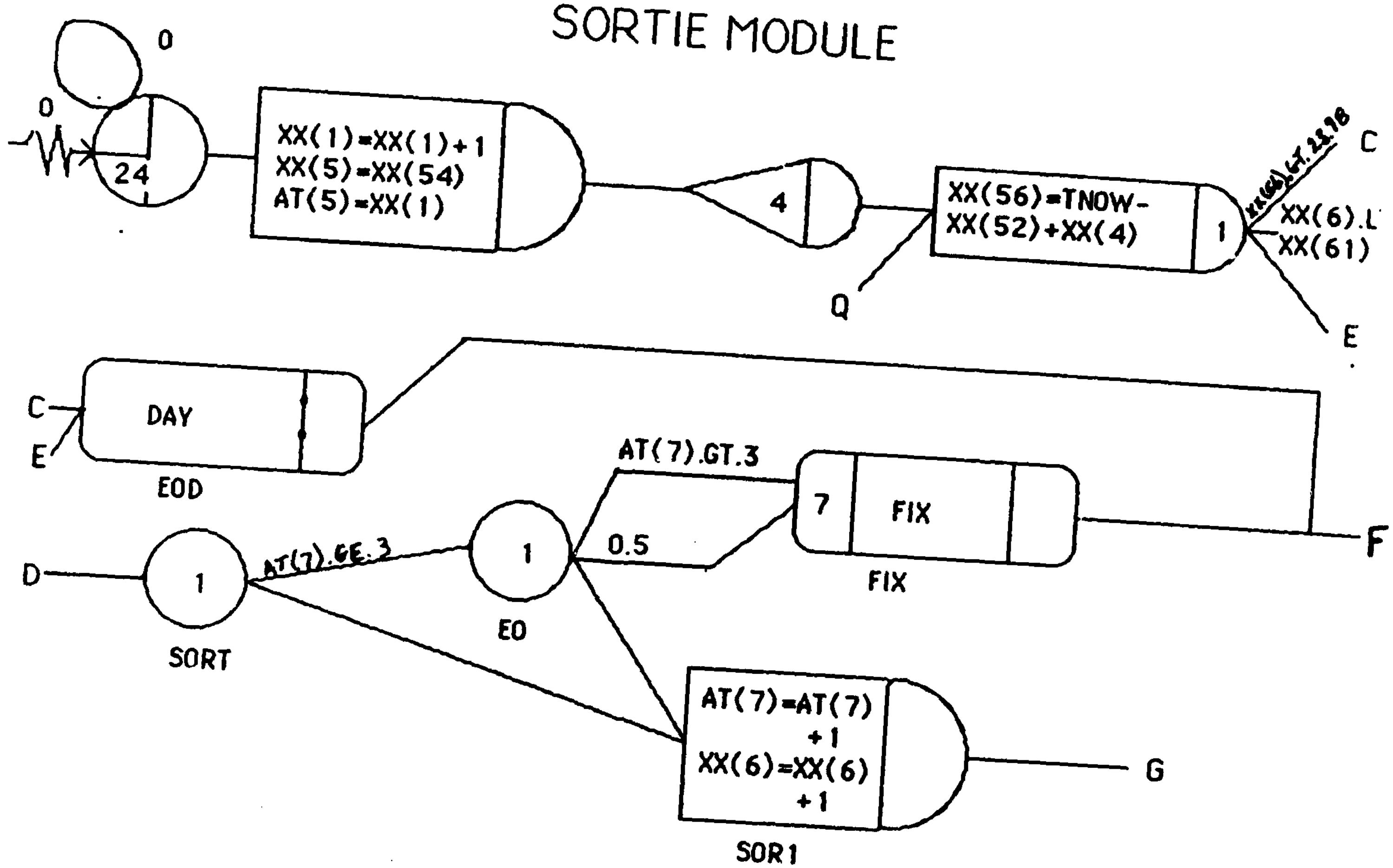
SOFTWARE NAME	BRIEF DESCRIPTION	COMPUTERS	OPERATING SYSTEM	OTHER SOFTWARE	RAM	OTHER HARDWARE	MODELING CAPABILITIES	APPROXIMATE PRICE
Simulations	A statistical software package that allows one to analyze and explore future possibilities. Consists of GASS and Monte Carlo Simulations	Apple II, II+, IIe and III (emulation) IBM PC	DOS 3.3 DOS 2.0		48K 128K	Printer req.		$395
SLAM II	An advanced FORTRAN based language that combines network, discrete event and continuous modeling capabilities. Models can be developed from the process-interaction, next-event, & activity-scanning perspectives	IBM PC & compatibles	MS DOS 2.x	Microsoft-FORTRAN 3.13, 3.2 (not req. to run network models)	320K	2 disk drives (1 req. for network models)	Discrete, continuous & combined	$975/$200 (quantity discounts & mult. site licenses avail.)
SOLON	An interactive graphics package for performance evaluation & selection of management plans in transportation operations	Apple II IBM PC	MS DOS 2.0	USCD Pascal	up to 256K		Discrete	$200-$800 (discounts for educ. inst.)
TUTSIM	An interactive graphics package for the simulation of continuous dynamic systems. Uses a block diagram format for constructing models	IBM PC, XT, AT, jr. Apple II, IIe, III CP/M80	MS DOS DOS 3.3		64K 48K 48K	IBM, Hercules or equiv. graphics adaptor	Continuous	$495 (for IBM vers.) $475 (for others) (quantity discounts & site licenses avail.)

Appendix B: Model Flow Diagrams and Model Code

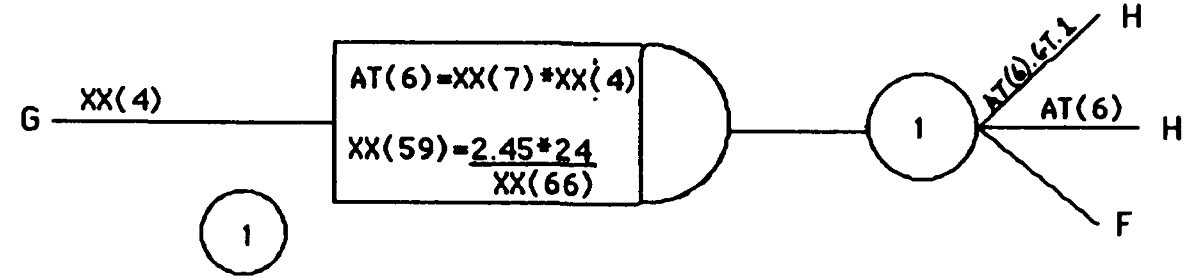
SCHEDULING MODULE



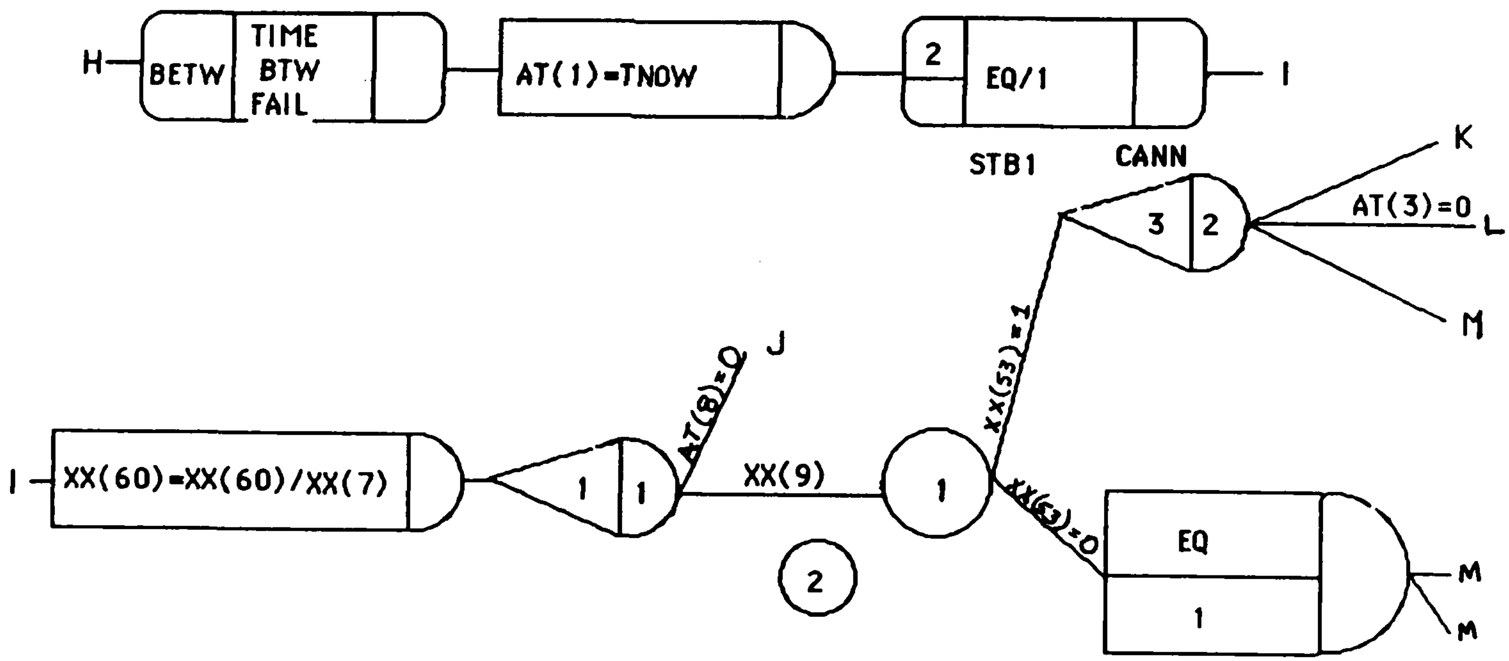
SORTIE MODULE



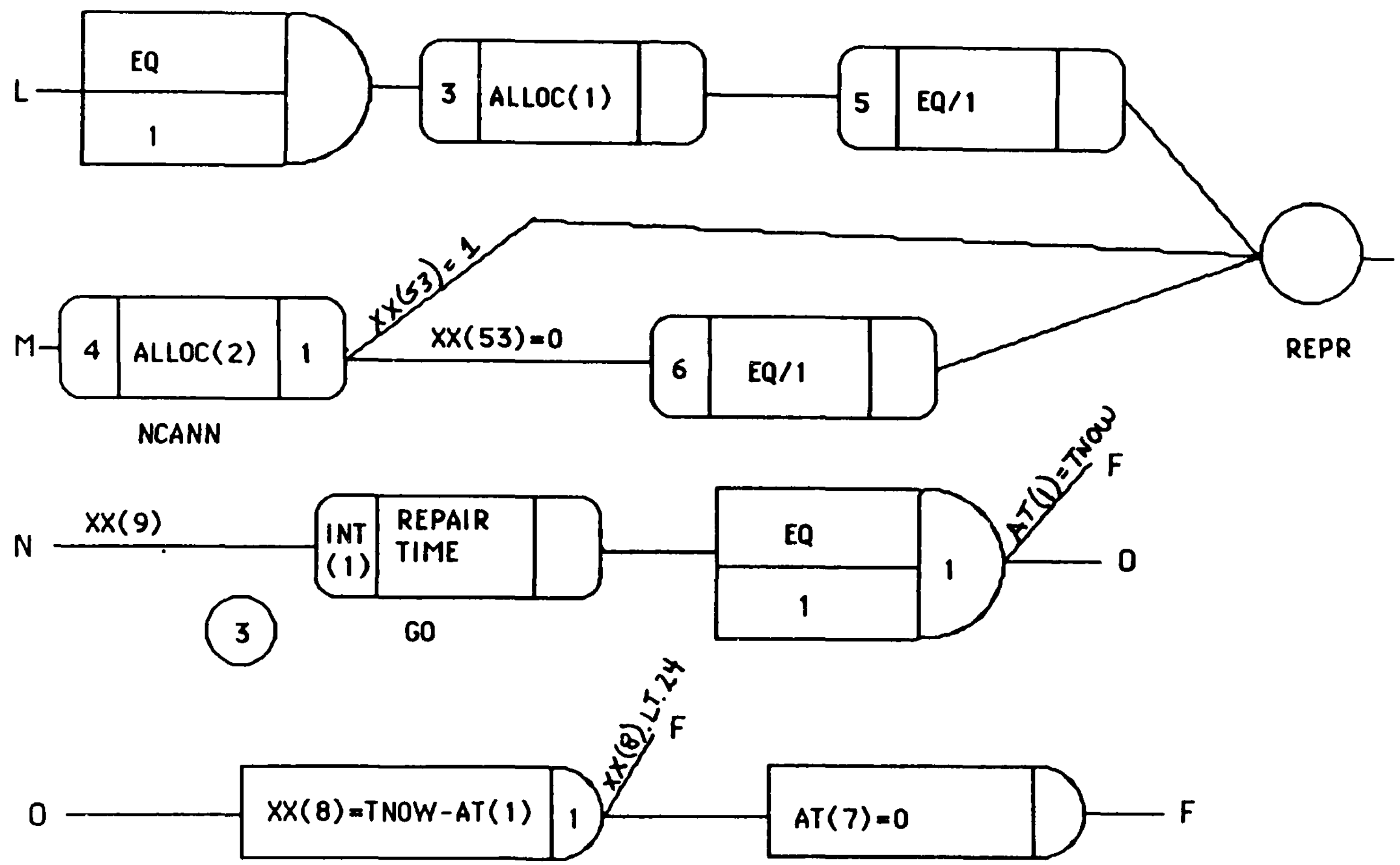
SORTIE MODULE



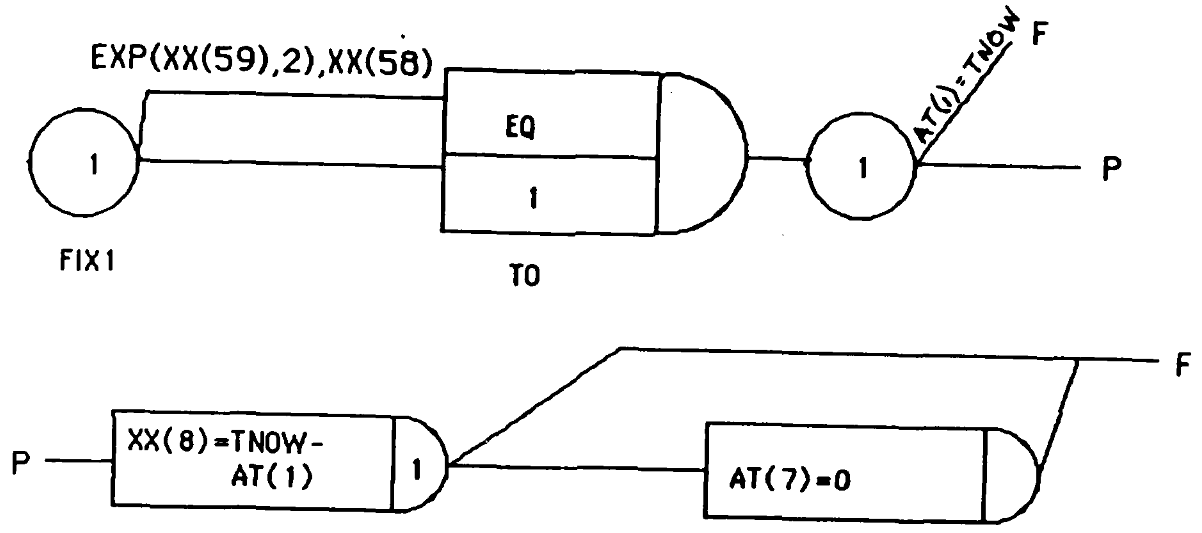
REPAIR MODULE



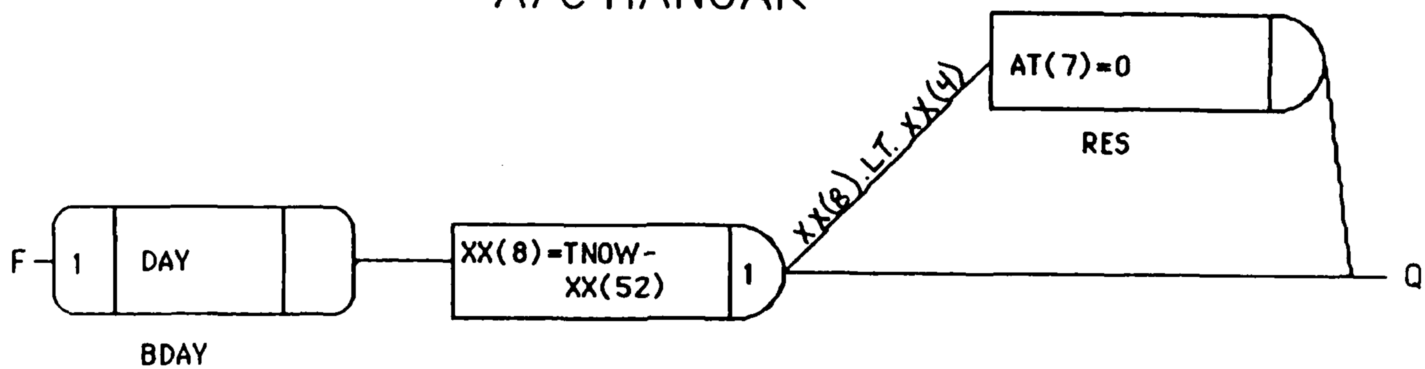
REPAIR MODULE

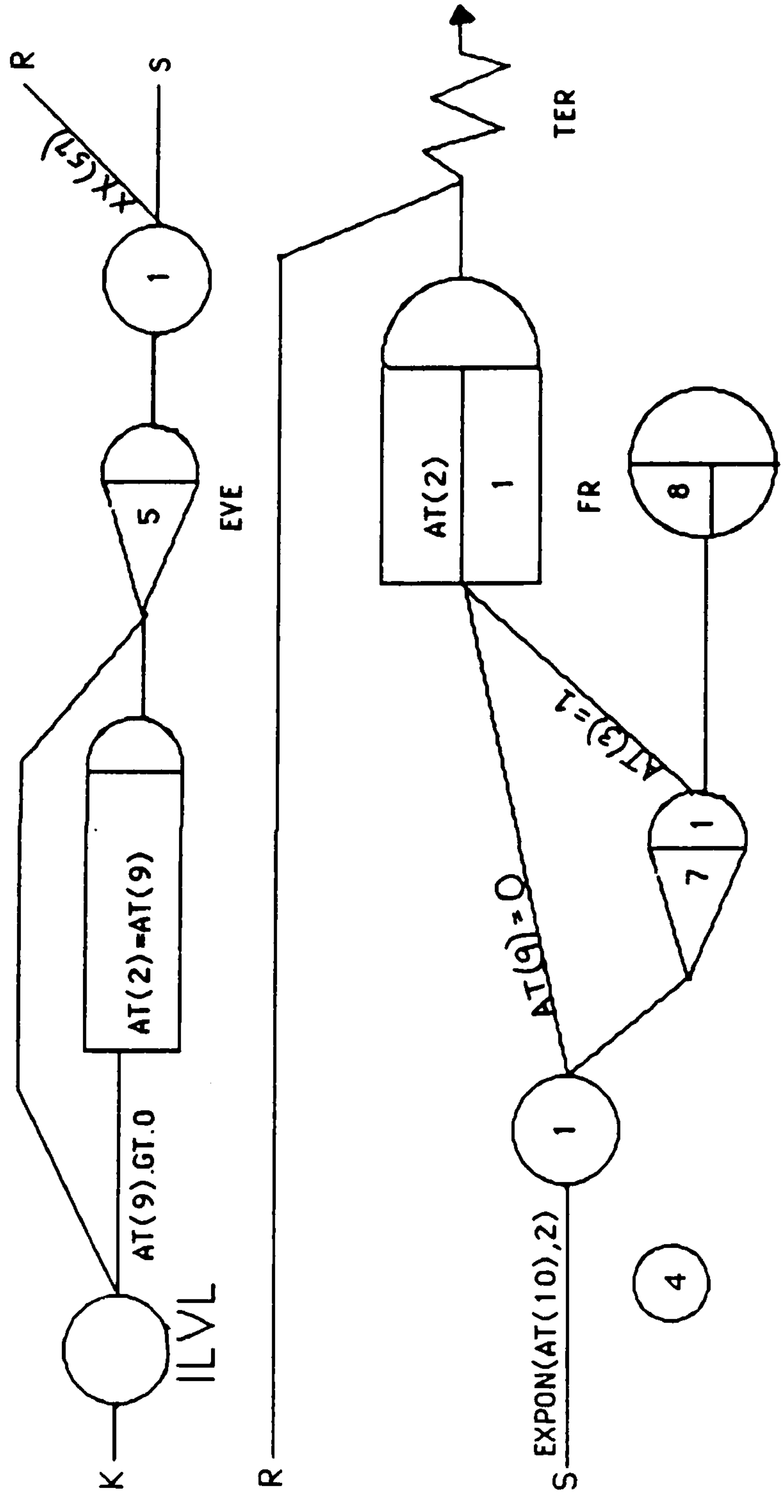


REPAIR MODULE



A/C HANGAR



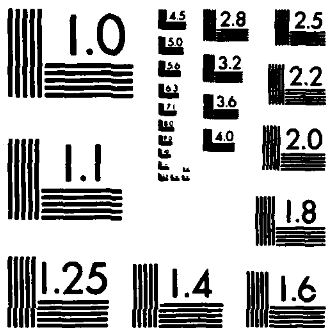

I-LVL MAINTENANCE
ILVL
K
AT(9).GT.0
AT(2)=AT(9)
5
EVE
1
XX(57)
R
S
R
AT(2)
1
FR
TER
8
AT(3)=1
AT(9)=0
7
1
S
EXPON(AT(10),2)
4

SLAM MODEL CODE

```
C
C   FIRST TWO LINES SET FILE SPACE RESTRICTIONS
C   AND NO. OF RUNS
C
GEN,TLEWIS,THESIS,7/21/87,50,,N,,N,N,72;
LIM,10,10,500;
NETWORK;
      RESOURCE/LRU1(0),4,3;
      RESOURCE/LRU2(0),4,3;
      RESOURCE/LRU3(0),4,3;     LRU SPARES AVAILABLE
      RESOURCE/LRU4(0),4,3;
      RESOURCE/LRU5(0),4,3;
      RESOURCE/LRU6(0),4,3;
      RESOURCE/LRU7(0),4,3;
      RESOURCE/LRU8(0),4,3;     LRU SPARES AVAILABLE
      RESOURCE/LRU9(0),4,3;
      RESOURCE/LRU10(0),4,3;
      RESOURCE/LRU11(0),4,3;
      RESOURCE/LRU12(0),4,3;
      RESOURCE/LRU13(0),4,3;     LRU SPARES AVAILABLE
      RESOURCE/LRU14(0),4,3;
      RESOURCE/LRU15(0),4,3;
      RESOURCE/LRU16(0),4,3;
      RESOURCE/LRU17(0),4,3;
      RESOURCE/LRU18(0),4,3;     LRU SPARES AVAILABLE
      RESOURCE/LRU19(0),4,3;
      RESOURCE/LRU20(0),4,3;
      RESOURCE/LRU21(0),4,3;
      RESOURCE/LRU22(0),4,3;
      RESOURCE/LRU23(0),4,3;     LRU SPARES AVAILABLE
      RESOURCE/LRU24(0),4,3;
      RESOURCE/LRU25(0),4,3;
      RESOURCE/LRU26(0),4,3;
      RESOURCE/LRU27(0),4,3;     LRU SPARES AVAILABLE
      RESOURCE/LRU28(0),4,3;
      RESOURCE/LRU29(0),4,3;
      RESOURCE/LRU30(0),4,3;
      RESOURCE/LRU31(0),4,3;
      RESOURCE/LRU32(0),4,3;     LRU SPARES AVAILABLE
      RESOURCE/LRU33(0),4,3;
      RESOURCE/LRU34(0),4,3;
      RESOURCE/LRU35(0),4,3;
      RESOURCE/LRU36(0),4,3;
      RESOURCE/LRU37(0),4,3;     LRU SPARES AVAILABLE
      RESOURCE/LRU38(0),4,3;
      RESOURCE/LRU39(0),4,3;
      RESOURCE/LRU40(0),4,3;
```

AD-A189 513 UNIT LEVEL WRSK (WAR READINESS SPARES KIT) ASSESSMENT AND SORTIE GENERATI (U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI T P LEWIS DEC 87 AFIT/GOR/ENS/87D-9

UNCLASSIFIED F/G 15/5 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
C       EQUIPMENT/MAINTENCE MEN AVAILABLE

        RESOURCE/EQ(25),6,5,2;

        GATE,DAY,,1;
        GATE,FIX,,7;
C
C       SCHEDULING MODULE
C
C       GETS DAILY SCHEDULE
C
        CREATE,24,0;
C       COUNTS THE DAY AND THE HOUR
        ASS,XX(52)=XX(51)*24,XX(51)=XX(51)+1;
        EVENT,2;   GETS DAILY SCHEDULE
C       COUNTS THE NUMBER OF FMC AIRCRAFT AT THE BEGINNING OF
C       DAY
        ASS,XX(6)=0,XX(2)=0,XX(5)=NNQ(1)+NNQ(7);
        OPEN,FIX;
        ACT,0.005;
        CLOSE,FIX;     CONTROLS THE LAUNCH OF
        OPEN,DAY;           AIRCRAFT
        ACT,23.99;
        GOON;
        CLOSE,DAY;
C       COLLECTS DAILY OUTPUT STATISTICS
        EVENT,6;
        COLCT,XX(6),SORTIES FLOWN;
 TE     TERM;
C
C       AIRCRAFT MODULE
C
        CREATE,0,0,,24;    CREATES  AIRCRAFT
C       COUNTS FMC A/C AND MARKS THE TAIL NO.
        ASS,XX(1)=XX(1)+1,XX(5)=XX(54),ATRIB(5)=XX(1);
        EVENT,4;  ASSEMBLES AIRCRAFT
 BEG    ASS,XX(56)=TNOW-XX(52)+XX(4),1;
C
C       IS THERE ENOUGH TIME TO FLY ANOTHER SORTIE ?
C       IF NO, GO TO THE END OF THE DAY
C       IF YES, HAVE ALL THE REQUIRED SORTIES BEEN FLOWN ?
C       IF YES, GO TO THE END OF THE DAY
C       IF NO, FLY A SORTIE
C
        ACT,,XX(56).GT.23.98,EOD;
        ACT,,XX(6).LT.XX(61),SORT;
        ACT;
C
C       END OF THE DAY
C       CLOSE HANGAR
C       WAIT FOR NEXT DAY
C
```

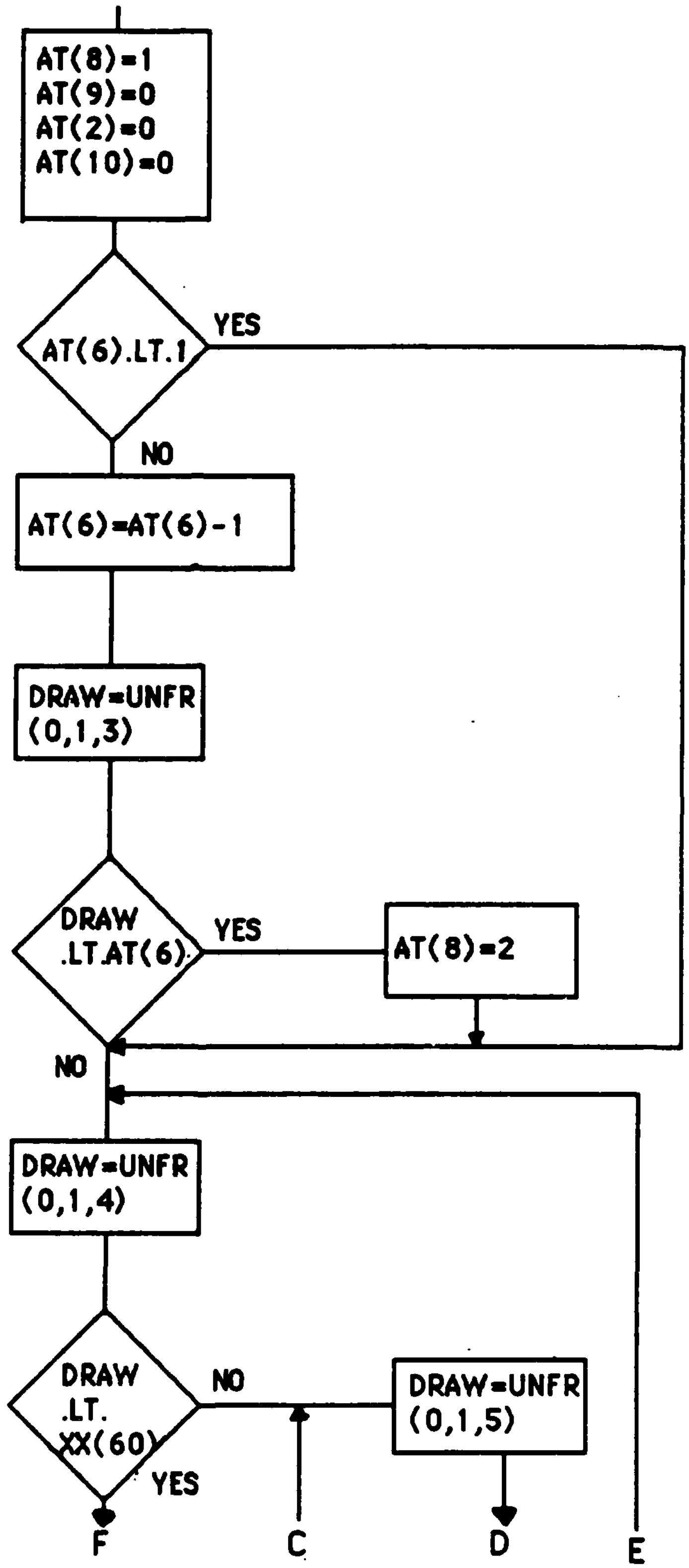
```
 EOD  CLOSE,DAY;
      ACT,,,BDAY;
C
C     FLY A SORTIE
C
C     IF A/C HAS FLOWN THE MAXIMUM SORTIE GENERATION RATE
C     SEND IT TO THE HANGAR
C
 SORT GOON,1;
      ACT,,ATRIB(7).GE.3,EO;
      ACT;
C     COUNTS SORTIES
 SOR1 ASS,ATRIB(7)=ATRIB(7)+1,XX(6)=XX(6)+1;
      ACT/1,XX(4); FLYS SORTIE
C
C     CALCULATES THE NUMBER OF LRU FAILURES AND THE AIRCRAFT
C     REPAIR TIME
C
      ASSIGN,ATRIB(6)=XX(7)*XX(4);
      ASSIGN,XX(59)=1.1142923*24/XX(66);
      GOON,1;
C
C     CHECKS TO SEE IF THE AIRCRAFT IS BROKEN
C     IF YES, GO TO THE REPAIR MODULE
C     IF NO, RETURN TO THE FLT. LINE
C
      ACT,,ATRIB(6).GT.1.0,BRK;
      ACT,,ATRIB(6),BRK;
      ACT,,,BDAY;
 EO   GOON,1;
      ACT,,ATRIB(7).GT.3,FIX;
      ACT,,0.5,FIX;
      ACT,,,SOR1;
C
C     HAVE ALL THE REQUIRED SORTIES BEEN FLOWN?
C     IF YES, GO TO HANGAR
C     IF NO, TRY TO FLY ANOTHER SORTIE
C
 BDAY AWAIT(1),DAY;
      ASS,XX(8)=TNOW-XX(52),1;
      ACT,,XX(8).LT.XX(4),RES;
      ACT,,,BEG;
 RES  ASS,ATRIB(7)=0;
      ACT,,,BEG;
C
C     REPAIR MODULE
C
C     COLLECTS TIME BETWEEN FAILURE
C
 BRK  COLCT,BETW,TIME BTW FAIL;
      ASSIGN,ATRIB(1)=TNOW; MARKS TIME OF FAILURE
 STB1 AWAIT(2),EQ/1;  WAITS FOR REPAIR MAN/EQUIPMENT
```

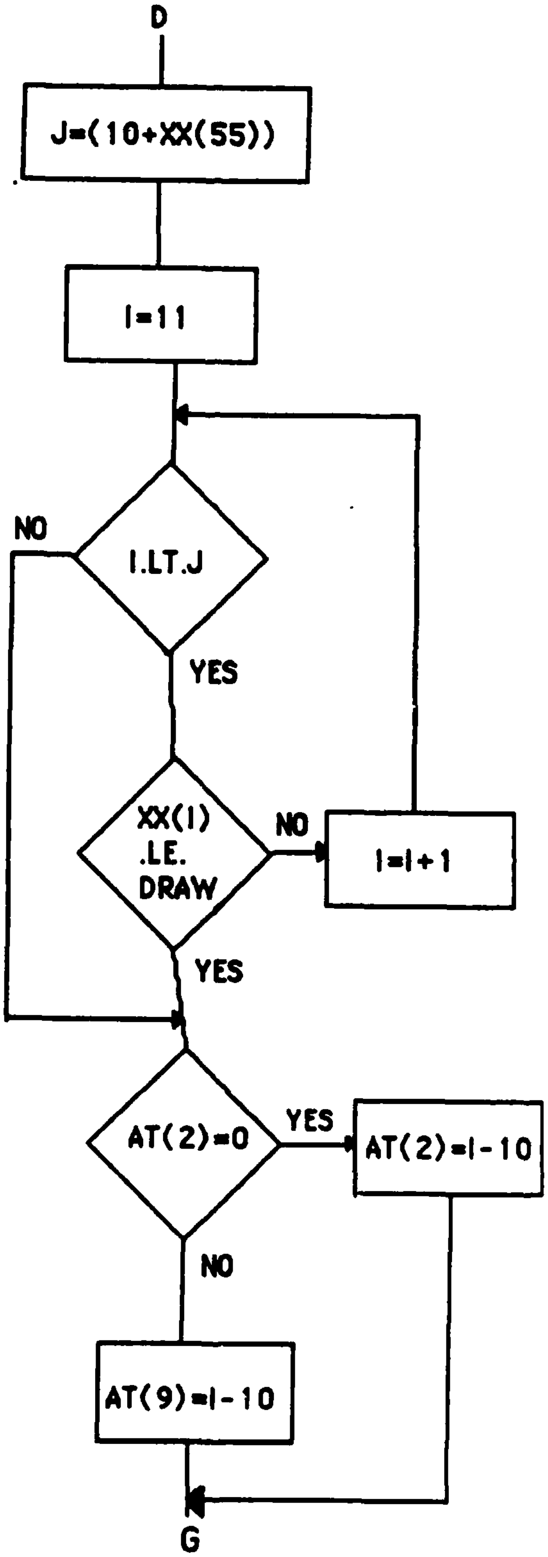
```
 STB  ASSIGN,XX(60)=XX(66)/XX(7),1;
C
C      CHOOSES BROKEN PART
C
       EVENT,1,1;
       ACT,,ATRIB(8).EQ.0,FIX1;
       ACT/2,XX(9);  REMOVES PART
C
C      CAN WE CANNIBALLIZE ?
C
       GOON,1;
       ACT,,XX(53).EQ.1,CANN;
       ACT,,XX(53).EQ.0;
       FREE,EQ/1;
       ACT,,,NCAN;
C
C      IF CANNIBALIZATION IS ALLOWED
C      DETERMINE IF LRU IS AVAILABLE?
C      IF YES, GET LRU
C      IF NO, MAKE A/C LRUS AVAILABLE TO OTHER A/C, WAIT FOR
C      THE BROKEN LRU AND RELEASE MAINTENENCE MAN/EQUIPMENT
C
C      SEND LRU TO REPAIR
C
 CANN EVENT,3,2;
       ACT,,,ILVL;
       ACT,,ATRIB(3).EQ.0,FRE;
       ACT;
C
C      IF CANNIBALIZATION IS NOT ALLOWED
C      DETERMINE IF LRU IS AVAILABLE?
C      IF YES, GET LRU
C      IF NO, WAIT FOR LRU AND RELEASE MAINTENENCE
C      MAN/EQUIPMENT
C
 NCAN AWAIT(4),ALLOC(2),1;
       ACT,,XX(53).EQ.1,REPR;
       ACT,,XX(53).EQ.0;
       AWAIT(6),EQ/1;
       ACT,,,REPR;
 FRE   FREE,EQ/1;    FREE MAINTENANCE MAN/EQUIPMENT
       AWAIT(3),ALLOC(1);  WAIT FOR LRU
       AWAIT(5),EQ/1;      WAIT FOR MAINTENANCE MAN/EQUIPMENT


 REPR GOON;
       ACT/3,XX(9);   REPLACE LRU
C
C      COLLECTS REPAIR TIME
C
 GO   COLCT,INT(1),REPAIR TIME;
C
```

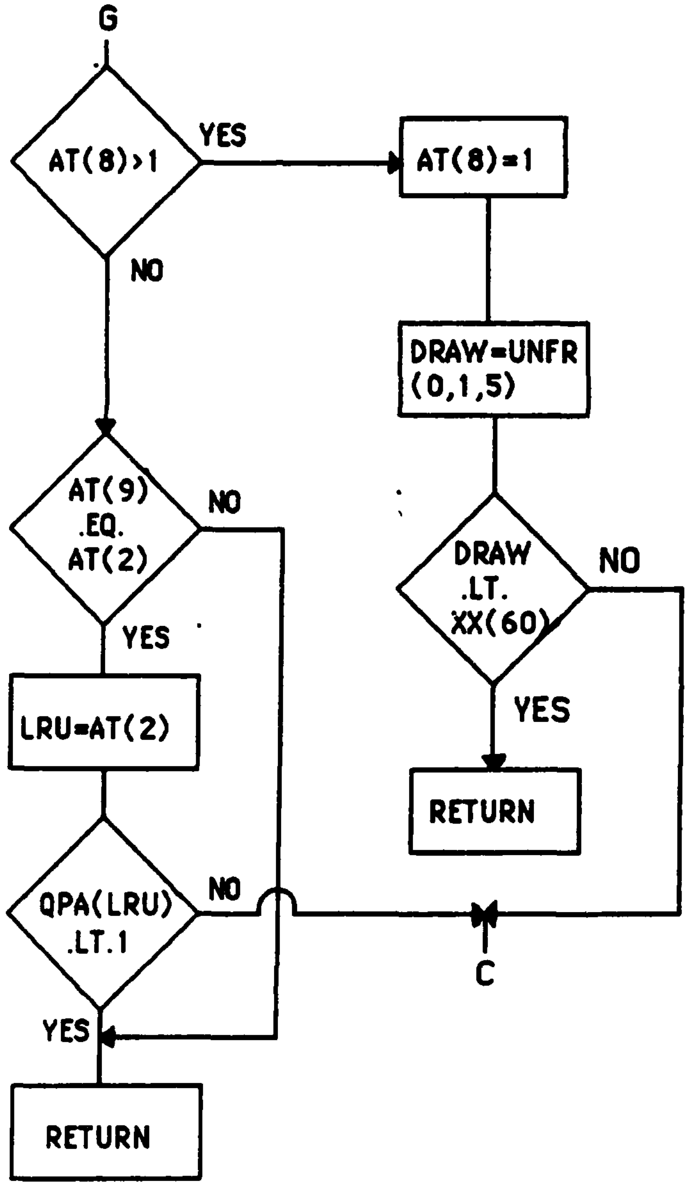
```
C      FREES MAINTENANCE RESOURCES
C
       FREE,EQ/1,1;
       ACT,,ATRIB(1).EQ.TNOW,BDAY;
       ACT;
C
C      RETURN TO FLIGHT LINE
C
       ASS,XX(8)=TNOW-ATRIB(1),1;
       ACT,,XX(8).LT.24,BDAY;
       ACT;
       ASS,ATRIB(7)=0;
       ACT,,,BDAY;
 FIX   AWAIT(7),FIX;
       ACT,,,BDAY;
C
C      A/C REPAIR CYCLE
C
 FIX1  GOON,1;
C
C      REPAIR TIME
C
       ACT,EXPON(XX(59),2),XX(58),TO;
       ACT;
C      FREE MAINTENANCE RESOURCES
  TO   FREE,EQ/1;
C
C      RETURN TO FLIGHT LINE
C
       GOON,1;
       ACT,,ATRIB(1).EQ.TNOW,BDAY;
       ACT;
       ASS,XX(8)=TNOW-ATRIB(1),1;
       ACT,,XX(8).LT.24,BDAY;
       ACT;
       ASS,ATRIB(7)=0;
       ACT,,,BDAY;
C
C      LRU REPAIR
C
 ILVL  GOON;
       ACT,,,EVE;
       ACT,,ATRIB(9).GT.0;
       ASS,ATRIB(2)=ATRIB(9);
C
C      CHOOSE BROKEN SRUS AND NRTS RATES
C
 EVE   EVENT,5;
       GOON,1;
       ACT,,XX(57),TER;
       ACT/4,EXPON(ATRIB(10),2);  REPAIR LRU
       GOON,1;
```

```
       ACT,,ATRIB(9).EQ.0,FR;
       ACT;
C
C      DECIDES IF SRU IS AVAILABLE?
C      IF YES, RETURNS LRU TO SUPPLY
C      IF NOT, THEN CANNIBALIZES SRUS, IF POSSIBLE
C
       EVENT,7,1;
       ACT,,ATRIB(3).EQ.1,FR;
       ACT;
       QUE(8);
 FR    FREE,ATRIB(2)/1;  FREE LRU
 TER   TERM;
       END;
INIT,0.,720,N;  RUNS MODEL FOR 30 DAYS (720 HRS.)
SIM;
FIN;
```

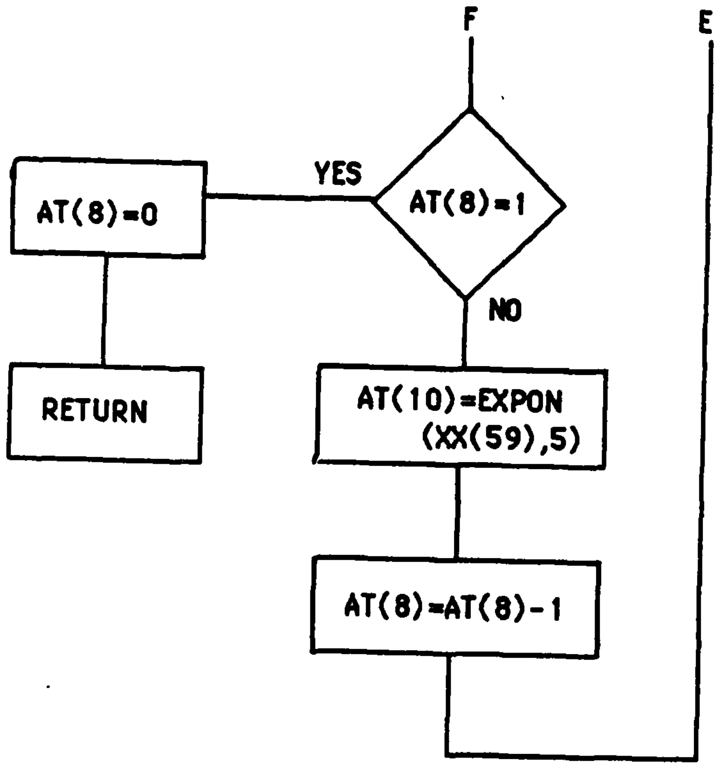
CHOOSES BROKEN LRUS



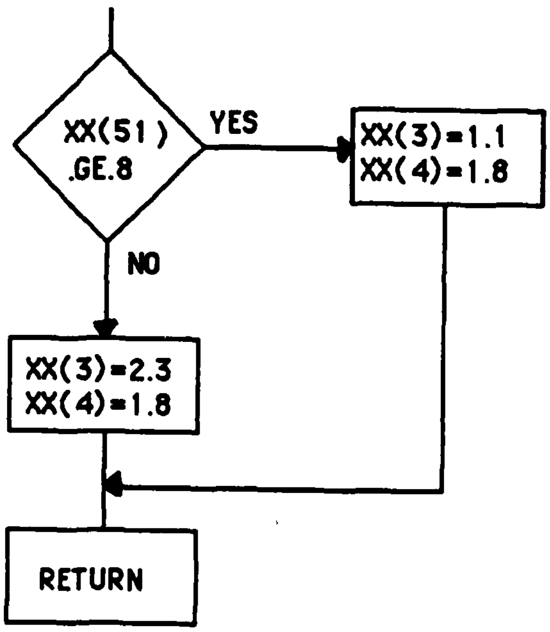

D
J=(10+XX(55))
I=11
I.LT.J
NO
YES
XX(I)
.LE.
DRAW
NO
I=I+1
YES
AT(2)=0
YES
AT(2)=I-10
NO
AT(9)=I-10
G

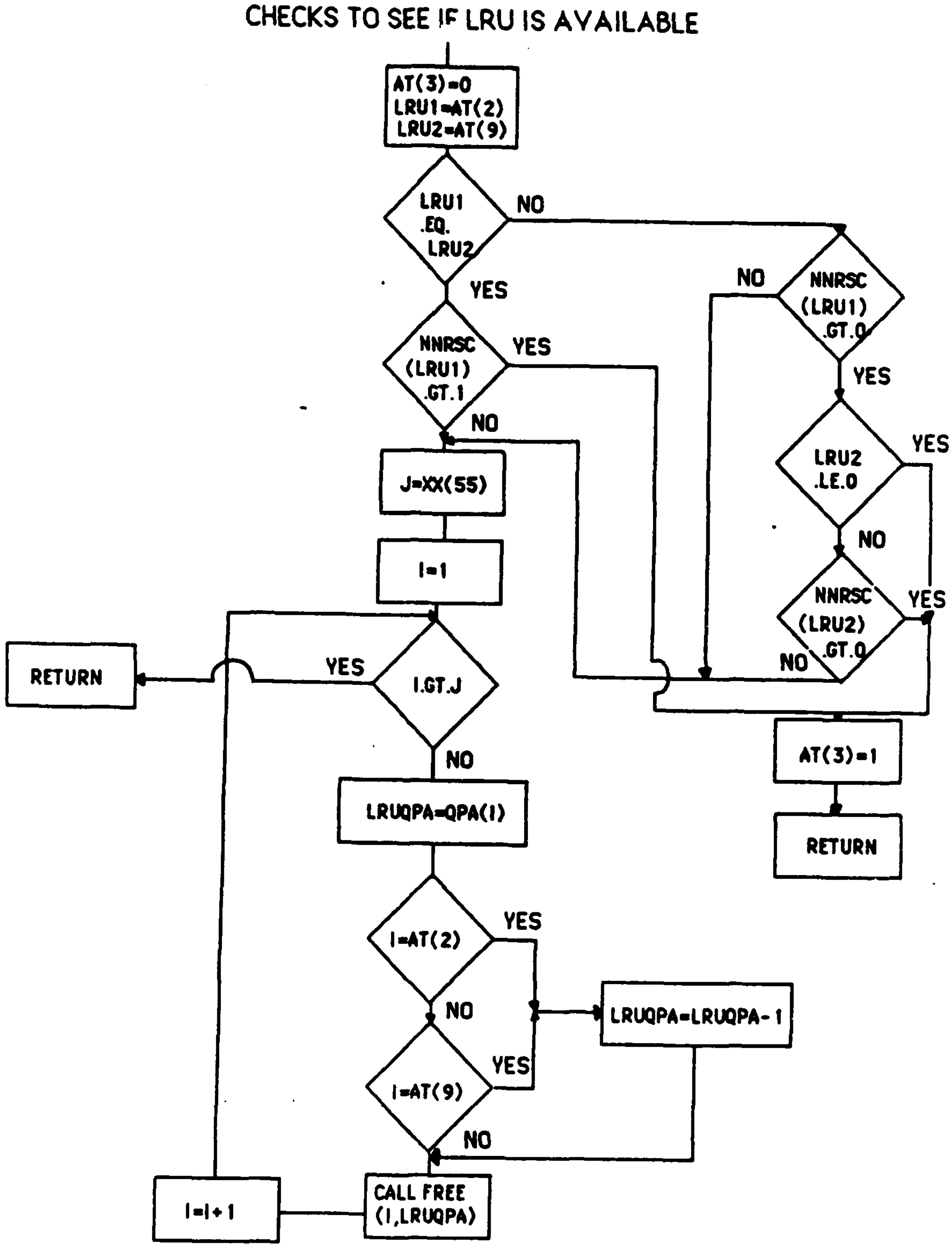


G
AT(8)>1
YES
NO
AT(8)=1
DRAW=UNFR (0,1,5)
AT(9) .EQ. AT(2)
NO
YES
DRAW .LT. XX(60)
NO
YES
LRU=AT(2)
RETURN
QPA(LRU) .LT.1
NO
C
YES
RETURN

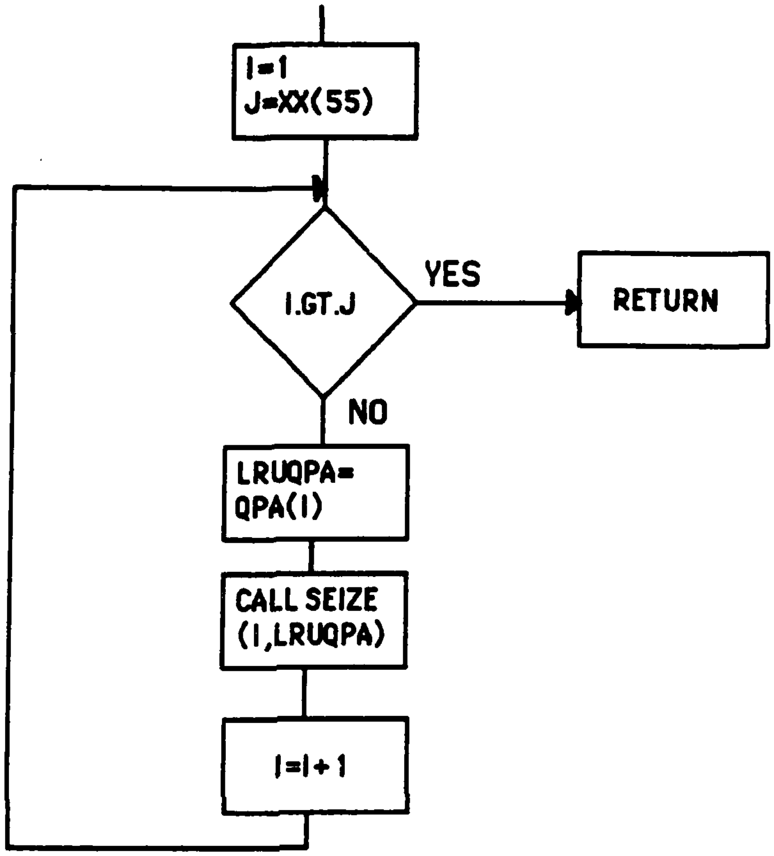


SETS FLIGHT SCHEDULE

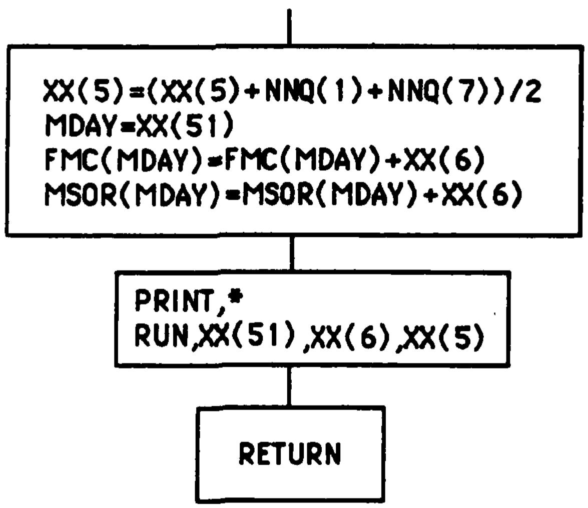



CHECKS TO SEE IF LRU IS AVAILABLE
AT(3)=0
LRU1=AT(2)
LRU2=AT(9)
LRU1 .EQ. LRU2
NO
YES
NNRSC (LRU1) .GT.1
YES
NO
NNRSC (LRU1) .GT.0
NO
YES
LRU2 .LE.0
YES
NO
NNRSC (LRU2) .GT.0
YES
NO
J=XX(55)
I=1
RETURN
YES
I.GT.J
NO
AT(3)=1
RETURN
LRUQPA=QPA(I)
I=AT(2)
YES
NO
LRUQPA=LRUQPA-1
YES
I=AT(9)
NO
I=I+1
CALL FREE (I,LRUQPA)

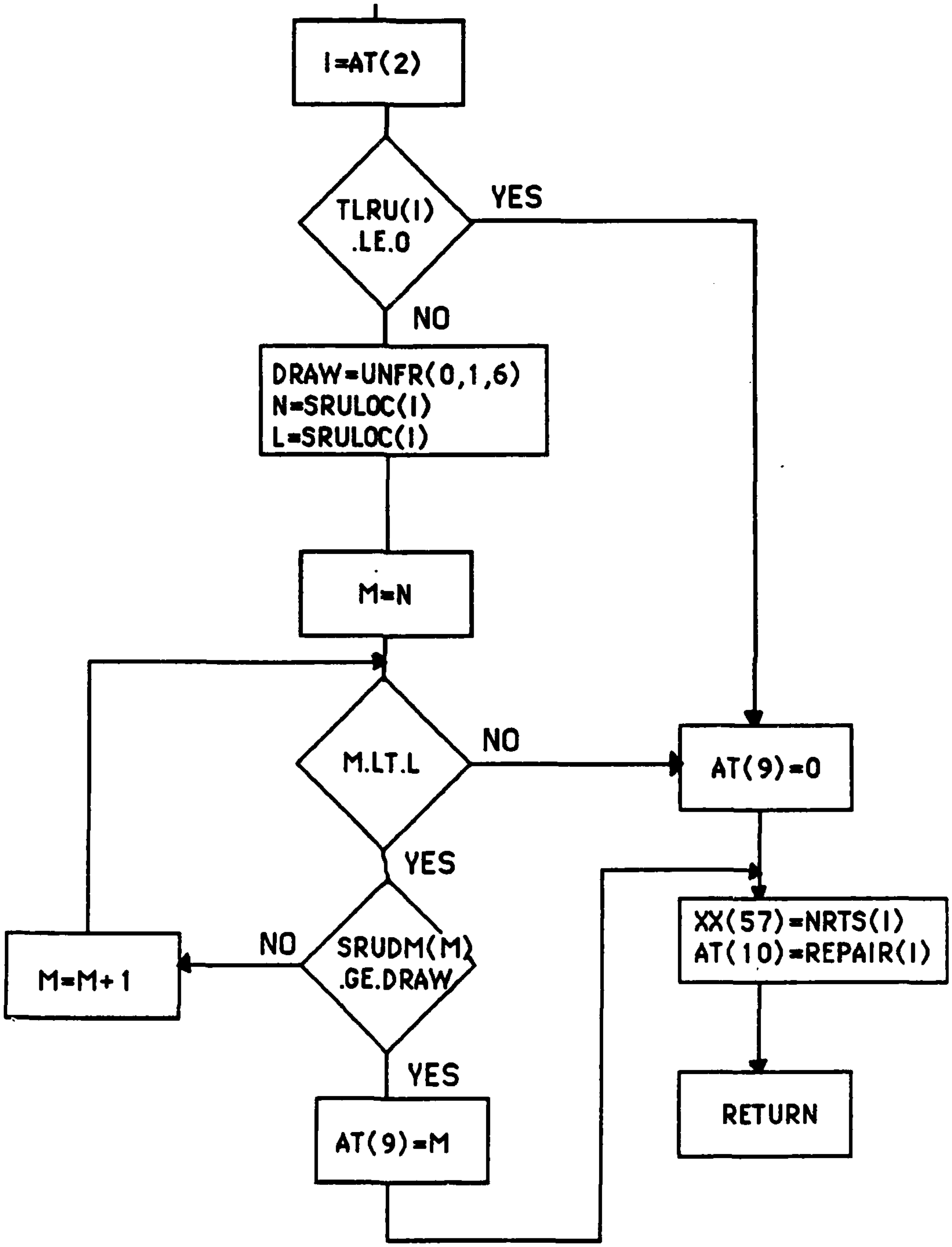
BUILDS AIRCRAFT



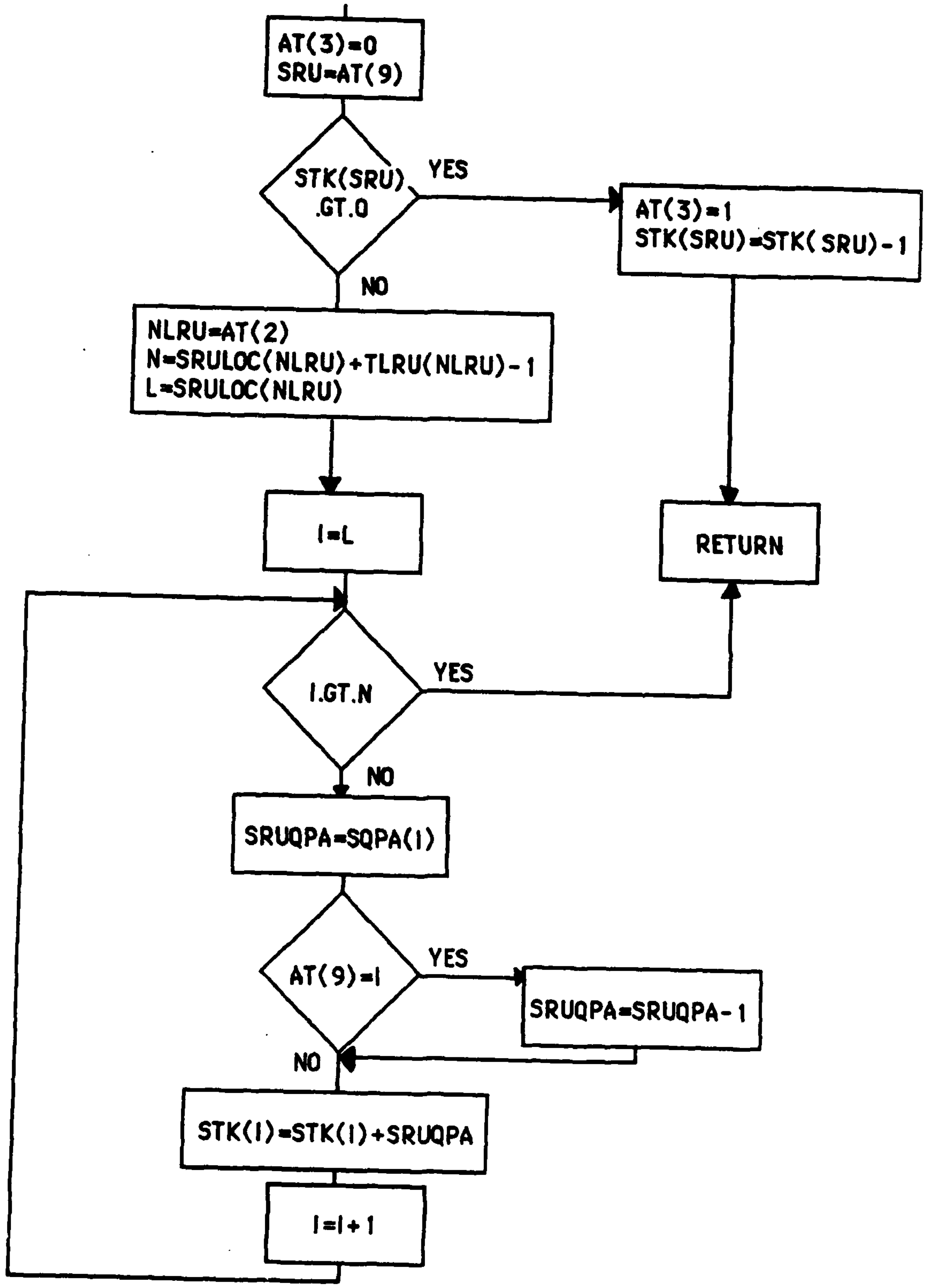
OUTPUT STATISTICS



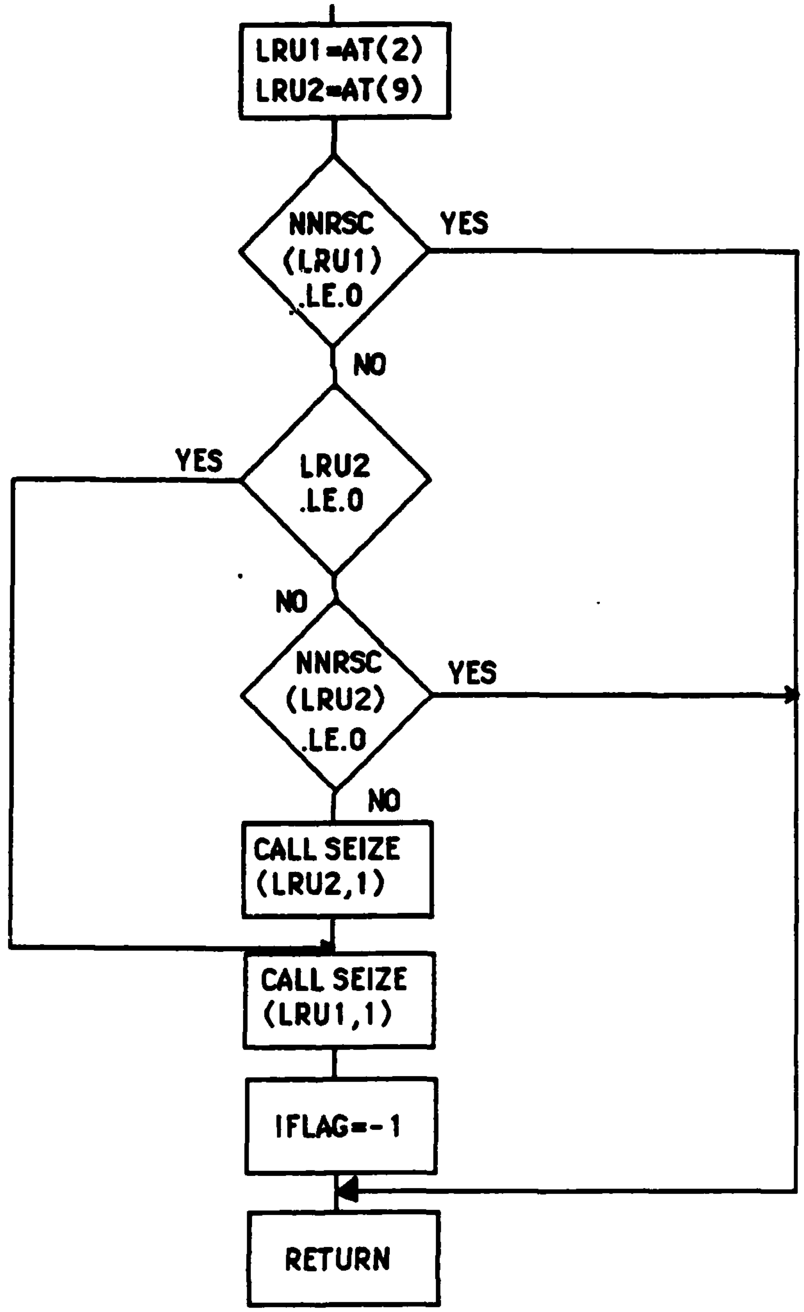
CHOOSES BROKEN SRU



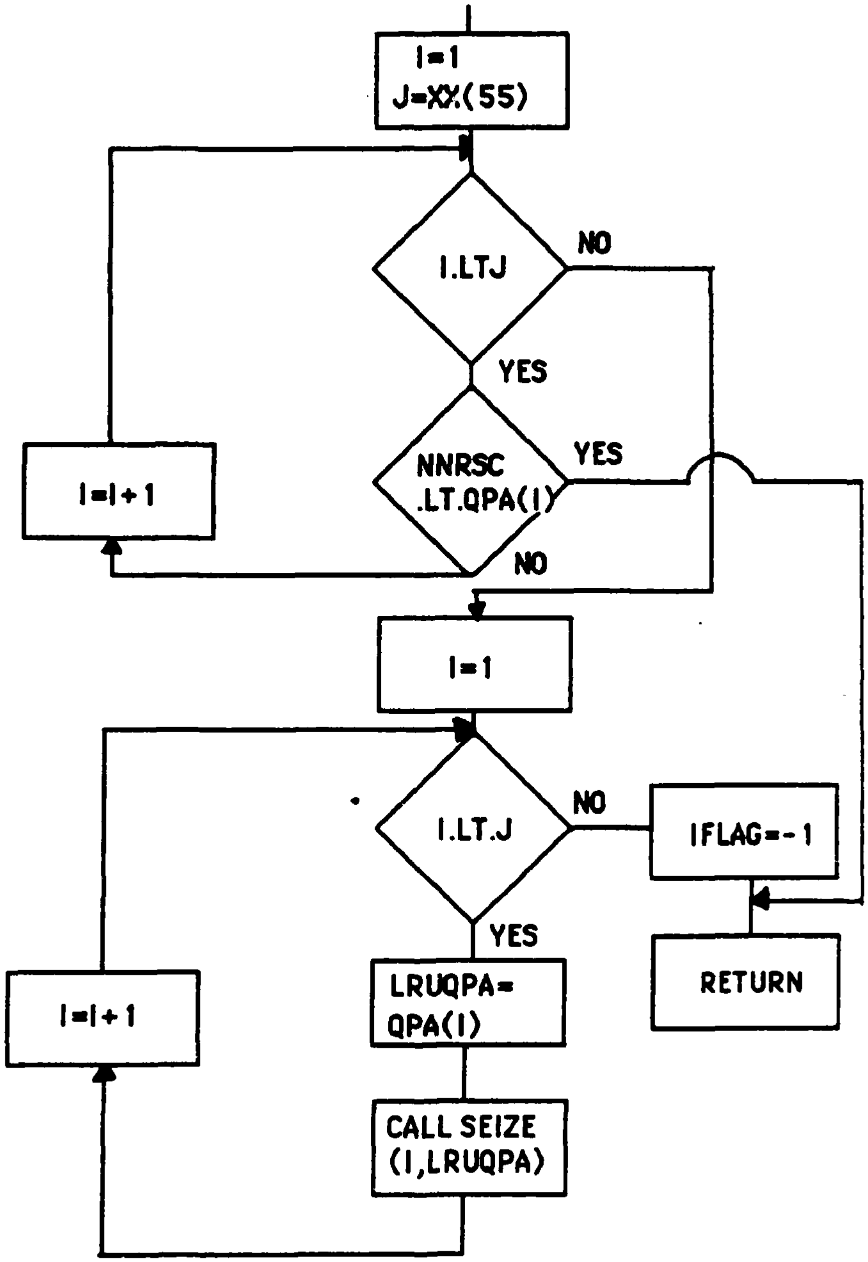
ALLOCATES SRUS



ALLOCATES LRUS (CANNS)



ALLOCATES LRUS (NO CANN)



SLAM FORTRAN EVENT CODE

```
INCLUDE:'PRCTL.FOR'
      PROGRAM MAIN
      DIMENSION NSET(10000)
COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,
NCLNR
1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,
XX(100)
      COMMON QSET(10000)
      EQUIVALENCE(NSET(1),QSET(1))
      NNSET=10000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      CALL SLAM
      END


      SUBROUTINE EVENT(I)

COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,
NCLNR
1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,
XX(100)
COMMON/MINE/REPAIRT(40),NRTS(40),QPA(40),RUN,STK(30),SQPA(30
)
1,TLRU(40),SRULOC(40),SRUDM(30),MSOR(30),FMC(30)
      GO TO (1,2,3,4,5,6,7) I
C
C     CHOOSES BROKEN LRU
C
 1    ATRIB(8)=1
      ATRIB(9)=0
      ATRIB(2)=0
      ATRIB(10)=0
C
C     USES RANDOM DRAWS TO CHOOSE BROKEN LRUS
C     AND THE NUMBER OF DEMANDS
C
      IF (ATRIB(6).LT.1) GO TO 17
      ATRIB(6)=ATRIB(6)-1
      DRAW=UNFRM(0.0,1.0,3)
      IF (DRAW.LT.ATRIB(6)) THEN
      ATRIB(8)=2
      END IF
 17   DRAW=UNFRM(0.0,1.0,4)
      IF (DRAW.LT.XX(60)) THEN
```

```
      IF (ATRIB(8).EQ.1.0) THEN
      ATRIB(8)=0
      GO TO 21
      END IF
C
C     GETS THE REPAIR TIME
C
      ATRIB(10)=EXPON(XX(59),5)
      ATRIB(8)=ATRIB(8)-1
      GO TO 17
      END IF
 15   DRAW=UNFRM(0.0,1.0,5)
      J=INT(10+XX(55))
      DO 10 I=11,J
      IF (XX(I).GE.DRAW) GO TO 20
 10   CONTINUE
 20   IF (ATRIB(2).EQ.0.0) THEN
C
C     MARKS THE BROKEN LRUS
C
      ATRIB(2)=I-10
      ELSE
      ATRIB(9)=I-10
      END IF
C
C     DECIDES WHERE THE REPAIR IS TO TAKE PLACE
C
      IF (ATRIB(8).GT.1.0) THEN
      ATRIB(8)=1
      DRAW=UNFRM(0.0,1.0,4)
      IF (DRAW.LT.XX(60)) GO TO 21
      GO TO 15
      END IF
      IF (ATRIB(9).EQ.ATRIB(2)) THEN
      LRU=INT(ATRIB(2))
      IF (QPA(LRU).GT.1) GO TO 21
      GO TO 15
      END IF
 21   RETURN

C
C     FLIGHT SCHEDULE
C
  2   IF(XX(51).GE.8) GO TO 22
      XX(3)=2.3       FIRST 7 DAYS
      XX(4)=1.8
      GO TO 24
 22   XX(3)=1.1       DAY 8 TO DAY 30
      XX(4)=1.8
C
C     GETS DAILY SORTIE REQUIREMENT
C
```

```
 24     XX(61)=XX(54)*XX(3)
        ROUND=XX(61)-INT(XX(61))
        IF (ROUND.LT.0.5) XX(61)=INT(XX(61))
        RETURN

C
C       ARE THE BROKEN LRUS AVAILABLE?
C

 3      ATRIB(3)=0
        LRU1=INT(ATRIB(2))
        LRU2=INT(ATRIB(9))
        IF (LRU1.EQ.LRU2) THEN
        IF (NNRSC(LRU1).GT.1) GO TO 30
        GO TO 35
        END IF
C
C       IF YES, GET LRUS
C
        IF (NNRSC(LRU1).GT.0) THEN
        IF (LRU2.LE.0) GO TO 30
        IF (NNRSC(LRU2).GT.0) GO TO 30
        END IF
 35     J=INT(XX(55))
C
C       IF THE BROKEN LRUS ARE NOT AVAILABLE, MAKE THE A/C
C       LRUS AVAILABLE TO OTHER AIRCRAFT
C
        DO 25 I=1,J
        LRUQPA=INT(QPA(I))
        IF (I.EQ.ATRIB(2)) LRUQPA=LRUQPA-1
        IF (I.EQ.ATRIB(9)) LRUQPA=LRUQPA-1
        CALL FREE(I,LRUQPA)
 25     CONTINUE
        RETURN
 30     ATRIB(3)=1
        RETURN
C
C       ASSEMBLES AIRCRAFT AT BEGINNING OF MODEL
C

 4      J=INT(XX(55))
        DO 40 I=1,J
        LRUQPA=INT(QPA(I))
        CALL SEIZE(I,LRUQPA)
 40     CONTINUE
        RETURN

C
C       DECIDES IF THERE IS A BROKEN SRU ?
C       IF YES, IT CHOOSES ONE
C
```

```
 5    I=INT(ATRIB(2))
      IF (TLRU(I).LE.0) GO TO 53

      DRAW=UNFRM(0.0,1.0,6)
      N=INT(SRULOC(I))
      L=INT(SRULOC(I)+TLRU(I)-1)
      DO 52 M=N,L
      IF(SRUDM(M).GE.DRAW) THEN
      ATRIB(9)=M
      GO TO 55
      END IF
  52  CONTINUE
  53  ATRIB(9)=0
C
C     GETS NRTS RATE AND REPAIR TIME
C
  55  XX(57)=NRTS(I)
      ATRIB(10)=REPAIRT(I)
      RETURN

C
C     CALCULATES OUTPUT STATISTICS
C
 6    XX(5)=(XX(5)+NNQ(1)+NNQ(7))/2
      MDAY=INT(XX(51))
      FMC(MDAY)=FMC(MDAY)+XX(5)
      MSOR(MDAY)=MSOR(MDAY)+XX(6)
      PRINT*,RUN,XX(51),XX(6),XX(5)
      RETURN

C
C     IS THE BROKEN SRU AVAILABLE ?
C     IF YES, FIX LRU
C     IF NO, CANNIBALIZE THE LRU, IF POSSIBLE
C
 7    ATRIB(3)=0
      SRU=INT(ATRIB(9))
      IF(STK(SRU).GT.0) THEN
      ATRIB(3)=1
      STK(SRU)=STK(SRU)-1
      GO TO 75
      END IF
      NLRU=INT(ATRIB(2))
      N=INT(SRULOC(NLRU)+TLRU(NLRU)-1)
      L=INT(SRULOC(NLRU))
      DO 72 I=L,N
      SRUQPA=INT(SQPA(I))
      IF(ATRIB(9).EQ.I) SRUQPA=SRUQPA-1
      STK(I)=STK(I)+SRUQPA
  72  CONTINUE
  75  RETURN
```

```
      END


      SUBROUTINE INTLC
COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,
NCLNR
1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,
XX(100)
COMMON/MINE/REPAIRT(40),NRTS(40),QPA(40),RUN,STK(30),SQPA(30
)
     1,TLRU(40),SRULOC(40),SRUDM(30)
      OPEN(UNIT=7,FILE='R2THESIS.INP',STATUS='OLD')
      OPEN(UNIT=8,FILE='SRU.INP',STATUS='OLD')

C
C     READS IN INITIAL VALUES AND FAILURE RATES
C
      RUN=RUN+1
      ISRU=0
      AM=0
      XX(58)=0.0125      A/C BREAK %
      XX(66)=0.32957     A/C DEMANDS/FLYING HOUR
      READ(7,*) AC
      XX(54)=AC         NO. OF A/C
      READ(7,*) BR
      XX(7)=BR           TOTAL DEMANDS/FLYING HOUR
      READ(7,*) CANN
      XX(53)=CANN        IS CANNIBALIZATION POSSIBLE ?
      READ(7,*) LRU
      XX(55)=LRU         NO. OF LRUS
      DO 10 I=1,LRU
C
C     READS IN LRU PART NO., DEMAND RATE, QPA, NO. OF LRUS
C     IN STOCK, NO. OF SRUS IN LRU, NRTS RATE, REPAIR CYCLE
C     TIME
C
      READ(7,*) NUM,FR,QPAS,LRUAV,SRUAV,RTS,REP
      TLRU(I)=SRUAV
      IF(SRUAV.EQ.0) GO TO 5
      ISRU1=ISRU+1
      ISRU=INT(ISRU+SRUAV)
      SRULOC(I)=ISRU1
      TOTAL=0
C
C     READS IN SRU DATA
C     SRU NO., DEMAND RATE, QPA, NO. OF SRUS IN STOCK
C
      DO 2 J=ISRU1,ISRU
      READ(8,*)NU,DM,QPSRU,SRUSTK
      SQPA(J)=QPSRU
```

```
      STK(J)=SRUSTK
      TOTAL=TOTAL+DM
      SRUDM(J)=TOTAL/FR
  2   CONTINUE
C
C     BUILDS THE CUMULATIVE FAILURE DISTRIBUTIONS
C
  5   NRTS(I)=RTS
      REPAIRT(I)=REP*24
      QPA(I)=QPAS
      LAV=INT(LRUAV+(XX(54)*QPA(I)))
      L=I+10
      XX(L)=AM+FR
      AM=XX(L)
      CALL ALTER(I,LAV)
 10   CONTINUE
      DO 20 I=1,LRU
      L=I+10
      M=10+INT(LRU)
      XX(L)=XX(L)/XX(M)
 20   CONTINUE
      CLOSE(7)
      CLOSE(8)
      RETURN
      END

      SUBROUTINE OTPUT

COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,
NCLNR
1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,
XX(100)
COMMON/MINE/REPAIRT(40),NRTS(40),QPA(40),RUN,STK(30),SQPA(30
)
     1,TLRU(40),SRULOC(40),SRUDM(30),MSOR(30),FMC(30)
      OPEN(UNIT=7,FILE='SAS.DAT',STATUS='NEW')

C
C     WRITES OUTPUT STATISTICS
C
      DO 10 I=1,30
      AVGFMC=FMC(I)/RUN
      AVGSOR=MSOR(I)/RUN
      IF (RUN.LT.50) GO TO 20
      WRITE(7,*)I,AVGFMC,AVGSOR
 10   CONTINUE

      CLOSE(7)
 20   RETURN
      END

      SUBROUTINE ALLOC(I,IFLAG)
```

```
COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,
NCLNR
1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,
XX(100)
      COMMON/MINE/REPAIRT(40),NRTS(40),QPA(40)

      IFLAG=0
      GO TO (1,2) I
C
C     CANNIBALIZATION ALLOWED
C
C     ARE LRUS AVAILABLE?
C     IF YES, GETS LRU
C     IF NO, USES FOR CANNIBALIZATION
C
 1    J=INT(XX(55))
      DO 10 I=1,J
      IF (NNRSC(I).LT.QPA(I)) GO TO 30
 10   CONTINUE
      DO 20 I=1,J
      LRUQPA=INT(QPA(I))
      CALL SEIZE(I,LRUQPA)
 20   CONTINUE
      IFLAG=-1
 30   RETURN

C
C     CANNIBALIZATION NOT ALLOWED
C
C     ARE LRUS AVAILABLE?
C     IF YES, GETS LRU
C     IF NO, WAITS FOR PARTS
C

 2    LRU1=INT(ATRIB(2))
      LRU2=INT(ATRIB(9))
      IF (NNRSC(LRU1).LE.0) GO TO 40
      IF (LRU2.LE.0) GO TO 35
      IF (NNRSC(LRU2).LE.0) GO TO 40
      CALL SEIZE(LRU2,1)
 35   CALL SEIZE(LRU1,1)
      IFLAG=-1
 40   RETURN
      END
```

Appendix C: Assumptions and Limitations

Model Assumptions and Limitations

- Demand Process is poisson.
- Demand process is independant of the repair process.
- The repair process uses times drawn from an exponential distribution with a given mean repair cycle time.
- Thirty day deployment.
- If cannibalization is possible, it will be instantaneous, include all parts, and always be successful.
- Capability to model 40 individual LRUs and 30 SRUs. The rest are combined into an overall aircraft failure rate.
- Maintenance is unconstrained.
- No resupply.
- No attrition.
- No more than 2 LRUfailures per sortie.
- The sum of the demand rates for a LRU's SRUs is less than or equal to the LRU demand rate.
- Selected LRU breaks take longer to fix than general aircraft breaks.

Appendix D: Input and Output Files

```
C
C   THIS IS THE DAILY EXPECTED VALUE OUTPUTS FOR THE
C   DYNA-METRIC TAC SAMPLE DATA.  IT GIVES THE DAY OF THE
C   DEPLOYMENT, THE NUMBER OF EXPECTED SORTIES FLOWN, AND
C   THE NUMBER OF FULLY MISSION-CAPABLE AIRCRAFT
C
```

DAY	SORTIE	FMC
1	55.2	21.5
2	54.94	18.91
3	53.83	17.16
4	51.6	15.61
5	48.5	14.24
6	45	13.01
7	41.47	11.91
8	25.97	12.2
9	26.04	12.39
10	26.08	12.61
11	26.11	12.54
12	26.12	12.51
13	26.13	12.41
14	26.12	12.25
15	26.10	12.03
16	26.06	11.75
17	25.99	11.42
18	25.89	11.03
19	25.72	10.60
20	25.46	10.12
21	25.09	9.59
22	24.56	9.01
23	23.84	8.40
24	22.88	7.76
25	21.67	7.08
26	20.18	6.39
27	18.44	5.68
28	16.49	4.96
29	14.37	4.25
30	12.18	3.55

```
C
C   THIS IS THE DAILY EXPECTED VALUE OUTPUTS FOR THE
C   DYNA-METRIC CORONET WARRIOR DATA.  IT GIVES THE DAY OF
C   THE DEPLOYMENT, THE NUMBER OF EXPECTED SORTIES FLOWN,
C   AND THE NUMBER OF FULLY MISSION-CAPABLE AIRCRAFT
C
```

DAY	SORTIE	FMC
1	55.2	22.05
2	55.15	20.03
3	55.19	20.51
4	55.18	20.21
5	55.16	19.84
6	55.12	19.47
7	55.07	19.12
8	26.4	19.35
9	26.4	19.43
10	26.4	19.43
11	26.4	19.38
12	26.4	19.29
13	26.4	19.19
14	26.4	19.07
15	26.4	18.95
16	26.4	18.81
17	26.4	18.67
18	26.4	18.53
19	26.4	18.38
20	26.4	18.23
21	26.4	18.07
22	26.4	17.91
23	26.4	17.74
24	26.4	17.56
25	26.4	17.37
26	26.4	17.16
27	26.4	16.94
28	26.39	16.69
29	26.39	16.42
30	26.38	16.11

C

C TAC SAMPLE INPUT DATA FILE FOR THE RESEARCH MODEL

C CREATED BY THE INTERACTIVE INPUT PROGRAM

C

24 - NO. OF AIRCRAFT

0.8154800 - AIRCRAFT DEMANDS/FLYING HOUR

1 - 1-CANNS ALLOWED/ 0-NO CANNS ALLOWED

35 - NO. OF LRUS MODELED

PART NO.	DEMAND RATE	LRU QPA	STOCK LVL	NO. OF SRU	NRTS	REPAIR TIME
3	3.8560000E-02	2	2	0	0.08	6.000000
18	2.7039999E-02	1	1	0	0.1	6.000000
16	0.02136000	1	3	0	0.09	6.000000
8	8.7800000E-03	2	1	0	1	6.000000
33	8.9499997E-03	1	3	0	1	2.000000
13	5.2800001E-03	2	4	0	1	6.000000
28	5.9099998E-03	1	0	0	0.36	4.000000
24	3.1900001E-03	1	0	0	1	3.000000
31	2.1660000E-02	2	13	0	1	6.000000
14	4.4200001E-03	2	0	0	1	6.000000
19	4.6100002E-03	1	0	0	0.04	4.000000
10	4.1800001E-03	2	1	0	1	6.000000
27	5.7500000E-03	1	2	0	0.21	4.000000
22	1.0950000E-02	1	7	0	1	3.000000
21	1.2040002E-02	2	7	0	1	6.000000
7	2.1899999E-03	1	0	0	1	3.000000
35	3.3400000E-03	2	0	0	1	6.000000
25	2.0200000E-03	1	0	0	1	5.000000
11	5.5200001E-03	1	3	0	1	5.000000
15	3.0600000E-03	2	0	0	1	6.000000
9	1.7900000E-03	1	0	0	1	3.000000
12	3.9400002E-03	1	7	0	0.1	4.000000
26	3.8400000E-03	2	1	0	1	6.000000
29	1.6100000E-03	1	0	0	1	6.000000
6	2.5800000E-03	2	0	0	1	6.000000
32	3.6299999E-03	1	3	0	1	6.000000
4	1.5000000E-03	1	0	0	1	3.000000
1	2.4800000E-03	2	5	0	1	6.000000
5	1.3980000E-02	1	12	0	1	5.000000
17	1.2700000E-03	1	0	0	1	4.000000
23	4.0899999E-03	1	0	0	0.1	3.000000

20	1.1300000E-03	1	0	0	1	3.000000
36	8.6000003E-03	1	3	0	0.01	5.000000
34	6.7500002E-03	1	2	0	0.15	4.000000
2	3.6800002E-03	4	0	0	1	6.000000

C

C CORONET WARRIOR INPUT DATA FILE FOR THE RESEARCH MODEL

C CREATED BY THE INTERACTIVE INPUT PROGRAM

C

24 - NO. OF AIRCRAFT

0.4649900 - AIRCRAFT DEMANDS/FLYING HOUR

1 - 1-CANNS ALLOWED/ 0-NO CANNS ALLOWED

35 - NO. OF LRUS MODELED

PART NO.	DEMAND RATE	LRU QPA	STOCK LVL	NO. OF SRU	NRTS	REPAIR TIME
1	1.2570000E-02	1.000000	0	0	0	1.800000
2	2.8599999E-03	1.000000	0	0	1	6.000000
3	1.3710000E-02	1.000000	3	0	0	5.060000
4	2.2900000E-03	1.000000	0	0	1	6.000000
5	4.5799999E-03	2.000000	1	0	1	6.000000
6	1.5420000E-02	2.000000	2	0	0	4.170000
7	8.5800001E-03	2.000000	0	0	0	0.8800000
8	1.7100000E-03	1.000000	0	0	1	5.000000
9	2.2799999E-03	2.000000	0	0	1	6.000000
10	2.2900000E-03	1.000000	0	0	0	0.9600000
11	8.0000004E-03	1.000000	0	0	0	0.5100000
12	1.7100000E-03	1.000000	1	0	1	6.000000
13	1.7100000E-03	1.000000	1	0	1	6.000000
14	6.2799999E-03	1.000000	1	0	0	3.210000
15	5.7199998E-03	4.000000	3	0	1	2.000000
16	4.5699999E-03	1.000000	3	0	0	9.880000
17	2.8599999E-03	2.000000	2	0	1	6.000000
18	5.6999997E-04	1.000000	0	0	1	5.000000
19	5.6999997E-04	1.000000	0	0	1	6.000000
20	5.6999997E-04	1.000000	0	0	1	6.000000
21	5.6999997E-04	1.000000	0	0	1	3.000000
22	5.6999997E-04	1.000000	0	0	1	6.000000
23	5.6999997E-04	1.000000	0	0	1	4.000000
24	5.6999997E-04	1.000000	0	0	1	2.000000
25	1.1399999E-03	2.000000	0	0	1	5.000000
26	1.1399999E-03	2.000000	0	0	1	6.000000
27	1.1399999E-03	2.000000	0	0	1	6.000000
28	3.4300000E-03	1.000000	4	0	1	6.000000
29	1.7199999E-03	2.000000	1	0	1	5.000000
30	1.7199999E-03	2.000000	1	0	1	6.000000

31	4.0000002E-03	1.000000	1	0	0	2.480000
32	2.8599999E-03	1.000000	4	0	1	6.000000
33	2.8599999E-03	1.000000	4	0	1	6.000000
34	9.1399997E-03	1.000000	3	0	0	1.910000
35	5.1400000E-03	1.000000	5	0	0	13.48000

Appendix E: Repair Times and Goodness of Fit Tests

```
C
C   REPAIR TIMES FOR PART NUMBER 6605010954108 IN HOURS
C

 57.300     61.500     56.800     53.300    108.500     32.800
103.100     84.900     75.300    161.000    164.000    179.500
179.800     33.900    167.000    245.500     69.000     37.000
 24.700     25.800     10.000     44.900     26.700     13.500
 77.600     11.500     15.600     19.200     32.500     57.200
 63.200     61.600

C
C   REPAIR TIMES FOR PART NUMBER 5865010668075 IN HOURS
C

 93.700     68.300    210.500     98.000    110.800      6.300
149.200      2.500      2.900     42.000    295.400      9.200
  9.100      6.200     72.000     32.800     97.800     76.700
 72.500     77.000    114.000     27.100    626.800

C
C   REPAIR TIMES FOR PART NUMBER 5841011007363 IN HOURS
C

39.000     39.200     43.300     32.700     20.000     18.000
71.000     13.900     88.400     25.200     14.800     24.800
 7.600      8.300     13.300    358.100      2.000      6.300
12.200     12.600     13.800

C
C   REPAIR TIMES FOR PART NUMBER 5841012348535 IN HOURS
C

 8.200     17.500     16.000     21.000      8.200      5.300
18.000     27.100     30.800      7.900     27.000     25.500
49.800     27.700     65.100     66.000      8.500     12.300
38.500     11.500      5.100     45.000    140.000     26.800
 8.000      4.000     48.000     47.000      7.300      7.200
 8.800      2.200     13.000     15.000      9.300     52.400
 5.500      6.100     11.100     11.500     11.500     23.800
 8.700      5.000     17.600

C
C   REPAIR TIMES FOR PART NUMBER 5841011356194 IN HOURS
C

31.000     35.300     44.500     32.800     10.800     17.600
18.400     22.000      5.000      9.000     14.300     33.900
```

11.000	18.000	27.800	21.900	19.900	21.200
15.000	22.300	15.100	212.300	46.800	9.800
10.900	16.800	44.500	9.200	46.000	41.800
25.000	6.100	20.200	14.200	12.500	7.800
5.400	3.900	9.800	42.700	12.500	15.900
6.300	10.800	4.100	4.100	11.800	11.000
7.300					

Goodness Of Fit Tests

K-S TEST

DATA SET	CRITICAL VALUE	TEST STATISTIC EXPONENTIAL	TEST STATISTIC LOGNORMAL
1	0.287	0.209	0.119
2	0.234	0.129	0.096
3	0.275	0.173	0.218
4	0.203	0.151	0.095
5	0.194	0.157	0.068

NOTE: ALPHA=0.05

CHI-SQUARED TEST

DATA SET	CRITICAL VALUE	TEST STATISTIC EXPONENTIAL	CRITICAL VALUE	TEST STATISTIC LOGNORMAL
1	5.991	5.857	3.841	2.048
2	9.488	7.375	7.815	5.125
3	5.991	3.957	3.841	5.000
4	14.067	11.200	12.592	6.000
5	14.067	16.571	12.592	6.286

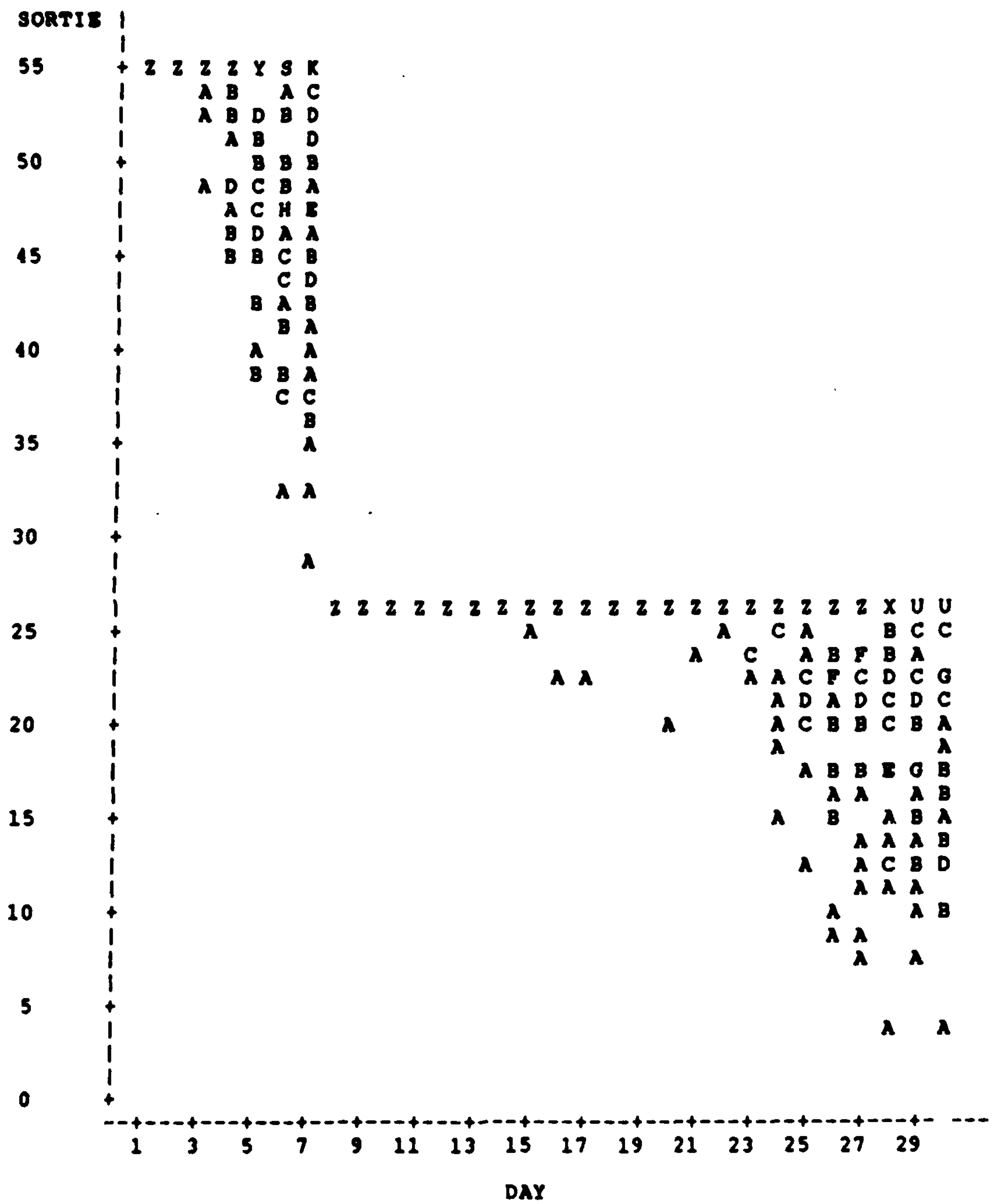
NOTE: ALPHA=0.05

Appendix F: Plots of Model Variances

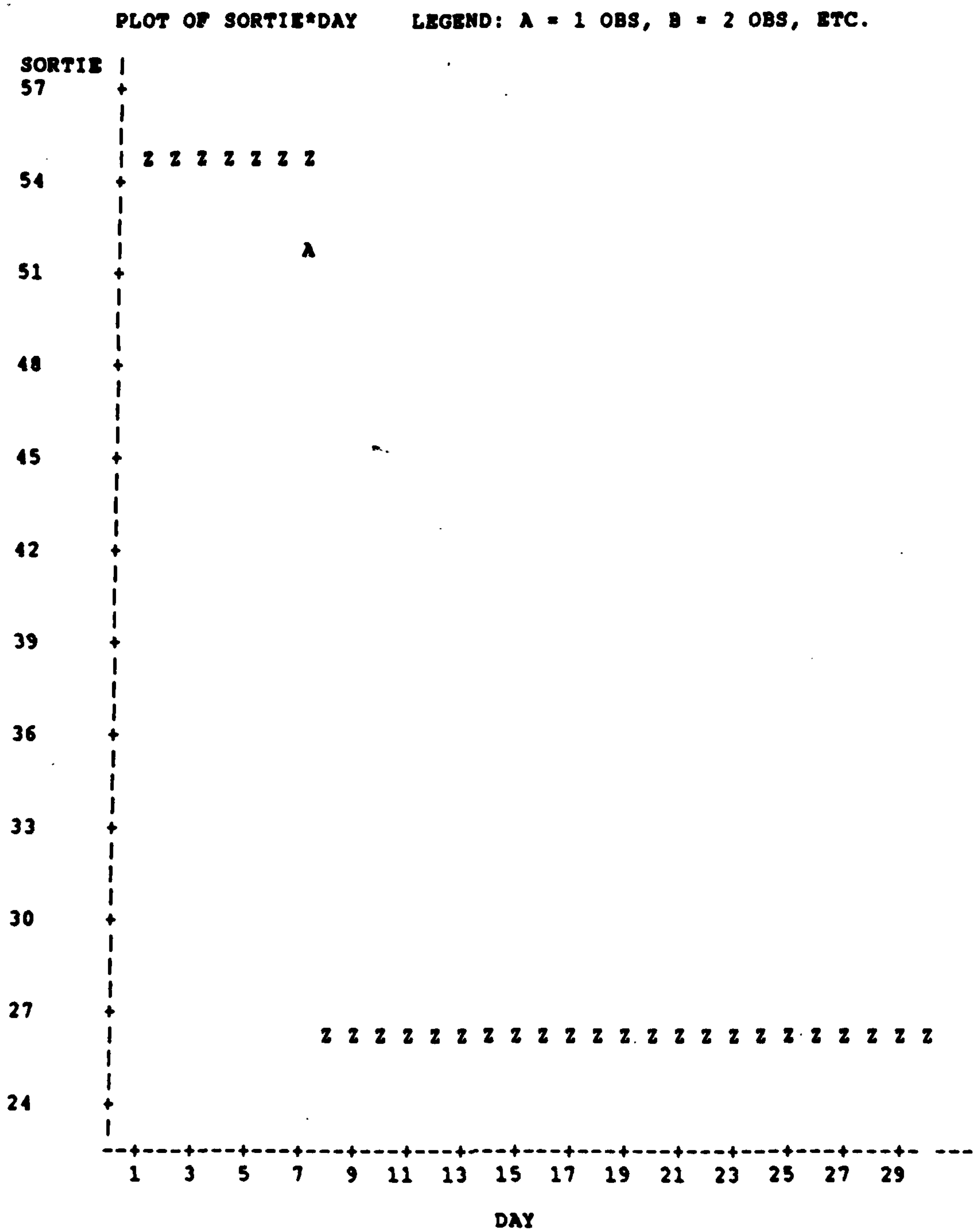
```
C

C   PROGRAM USED TO CALCULATE MEANS

C   AND PLOT THE RESULTS (SAS PROGRAM)

C

options ls=80;
FILENAME NEW'TEST3A.DAT';
data new;
infile new;
input RUN DAY SORTIE FMC;
proc SORT;
  BY DAY;
PROC MEANS;
  BY DAY;
  VAR SORTIE FMC;
PROC PLOT;
  PLOT SORTIE*DAY FMC*DAY;
```

VARIANCE PLOT OF 50 RUNS TAC SAMPLE DATA

PLOT OF SORTIE*DAY   LEGEND: A = 1 OBS, B = 2 OBS, ETC.

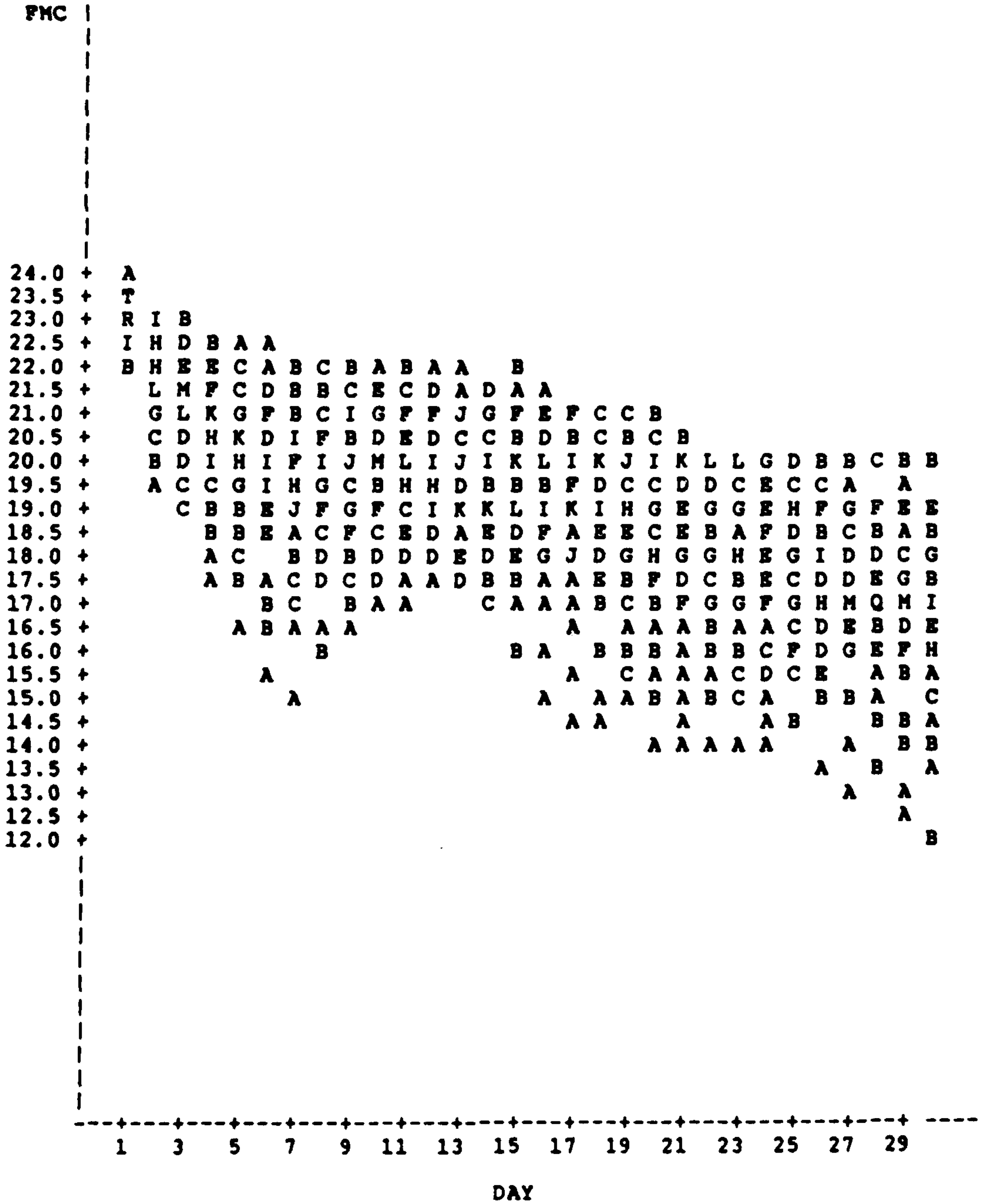


VARIANCE PLOT OF 50 RUNS CORONET WARRIOR DATA



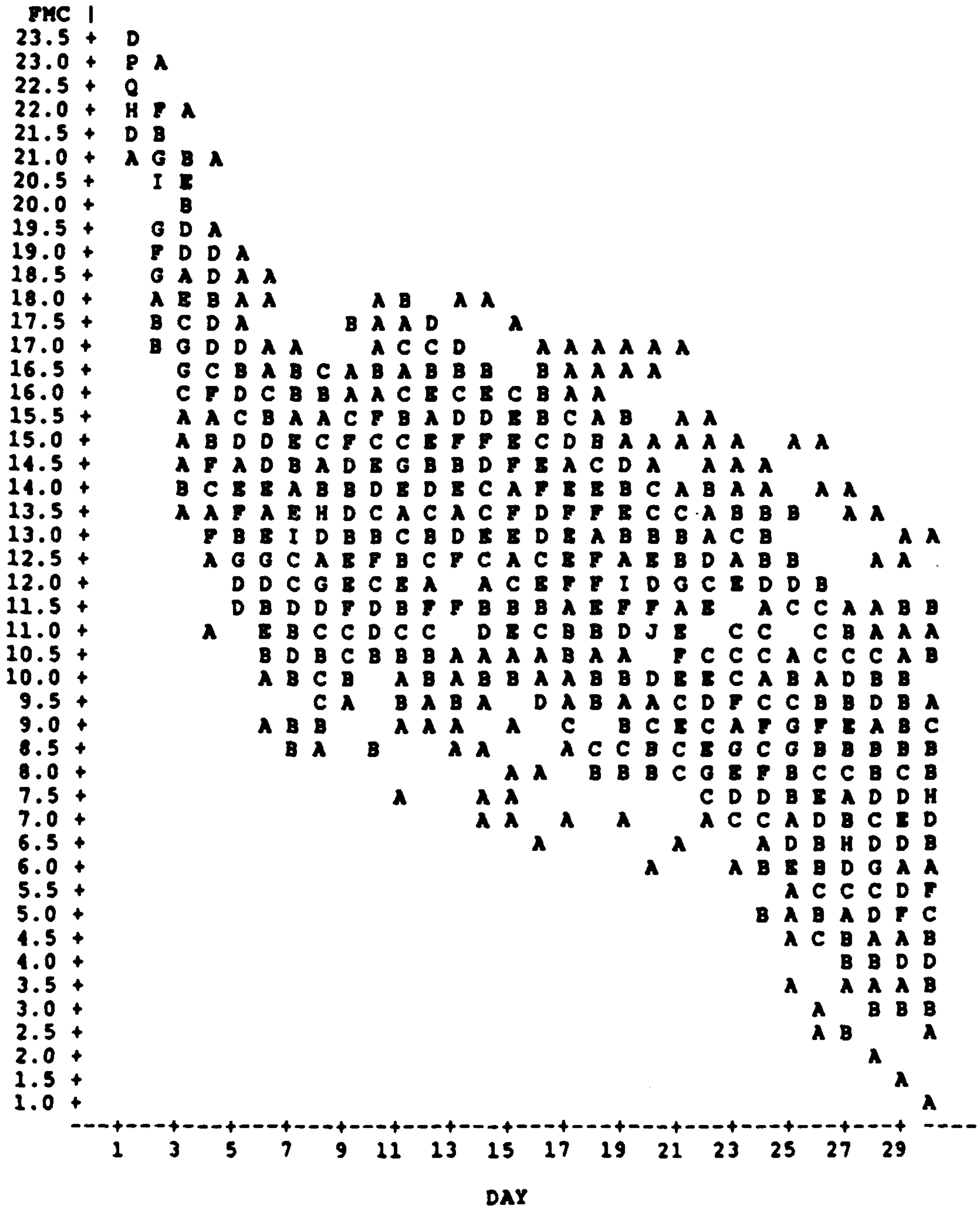
VARIANCE PLOT OF 50 RUNS TAC SAMPLE DATA

PLOT OF FMC*DAY   LEGEND: A = 1 OBS, B = 2 OBS, ETC.



VARIANCE PLOT OF 50 RUNS CORONET WARRIOR DATA

PLOT OF FMC*DAY   LEGEND: A = 1 OBS, B = 2 OBS, ETC.



Appendix G: Research Results

DAILY OUTPUT FOR

TAC SAMPLE DATA

	RESEARCH MODEL		DYNA-METRIC MODEL	
DAY	FMC	SORTIE	FMC	SORTIE
1	22.55	55.00	21.50	55.20
2	19.93	55.00	18.91	54.94
3	17.81	54.80	17.16	53.83
4	15.99	53.38	15.61	51.60
5	14.35	51.10	14.24	48.50
6	13.38	48.82	13.01	45.00
7	12.77	47.34	11.91	41.47
8	12.52	26.00	12.20	25.97
9	13.11	26.00	12.39	26.04
10	13.47	26.00	12.61	26.08
11	13.53	26.00	12.54	26.11
12	13.74	26.00	12.51	26.12
13	13.69	26.00	12.41	26.13
14	13.36	26.00	12.25	26.12
15	13.07	25.98	12.03	26.10
16	12.94	25.94	11.75	26.06
17	12.75	25.94	11.42	25.99
18	12.40	26.00	11.03	25.89
19	11.98	26.00	10.60	25.72
20	11.53	25.88	10.12	25.46
21	10.92	25.96	9.59	25.09
22	10.35	25.98	9.01	24.56
23	9.98	25.82	8.40	23.84
24	9.52	25.30	7.76	22.88
25	8.87	24.58	7.08	21.67
26	8.24	23.56	6.39	20.18
27	7.64	22.88	5.68	18.44
28	7.19	22.10	4.96	16.49
29	6.97	21.42	4.25	14.37
30	6.68	21.32	3.55	12.18

DAILY OUTPUT FOR

CORONET WARRIOR DATA

	RESEARCH MODEL		DYNA-METRIC MODEL	
DAY	FMC	SORTIE	FMC	SORTIE
1	23.09	55.00	22.05	55.20
2	21.78	55.00	20.03	55.15
3	21.10	55.00	20.51	55.19
4	20.56	55.00	20.21	55.18
5	20.05	55.00	19.84	55.16
6	19.63	55.00	19.47	55.12
7	19.38	54.94	19.12	55.07
8	19.43	26.00	19.35	26.40
9	19.60	26.00	19.43	26.40
10	19.72	26.00	19.43	26.40
11	19.78	26.00	19.38	26.40
12	19.73	26.00	19.29	26.40
13	19.61	26.00	19.19	26.40
14	19.46	26.00	19.07	26.40
15	19.35	26.00	18.95	26.40
16	19.22	26.00	18.81	26.40
17	19.05	26.00	18.67	26.40
18	18.82	26.00	18.53	26.40
19	18.58	26.00	18.38	26.40
20	18.43	26.00	18.23	26.40
21	18.32	26.00	18.07	26.40
22	18.20	26.00	17.91	26.40
23	18.04	26.00	17.74	26.40
24	17.86	26.00	17.56	26.40
25	17.66	26.00	17.37	26.40
26	17.37	26.00	17.16	26.40
27	17.25	26.00	16.94	26.40
28	17.18	26.00	16.69	26.39
29	16.92	26.00	16.42	26.39
30	16.74	26.00	16.11	26.38

```
C   PROGRAM THAT COMPUTED THE DIFFERENCES BETWEEN THE
C   RESEARCH MODEL AND THE DYNA-METRIC MODEL
C   (FORTRAN PROGRAM)
C

      PROGRAM MAIN
C     OPENS DATA FILES
      OPEN(UNIT=7,FILE='TEST3A.DAT',STATUS='OLD')
      OPEN(UNIT=8,FILE='DYNA2.DAT',STATUS='OLD')
      OPEN(UNIT=9,FILE='DIFF.DAT',STATUS='NEW')

      DO 10 I=1,30
C     READS IN DATA
      READ(7,*)DAY,FMC,SORT
      READ(8,*)DAY1,SORT1,FMC1
C     CALCULATES DIFFERENCES
      DIFF=SORT-SORT1
      WRITE(9,*)DIFF
      DIFF=FMC-FMC1
      WRITE(9,*)DIFF
 10   CONTINUE
      CLOSE(7)
      CLOSE(8)
      CLOSE(9)
      END
```

C DIFFERENCE OUTPUT FROM TAC SAMPLE

DAY	FMC	SORTIES
1	1.05	-0.20
2	1.02	0.06
3	0.65	0.97
4	0.38	1.78
5	0.11	2.60
6	0.37	3.82
7	0.86	5.87
8	0.32	0.03
9	0.72	-0.04
10	0.86	-0.08
11	0.99	-0.11
12	1.23	-0.12
13	1.28	-0.13
14	1.11	-0.12
15	1.04	-0.12
16	1.19	-0.12
17	1.33	-0.05
18	1.37	0.11
19	1.38	0.28
20	1.41	0.42
21	1.33	0.87
22	1.34	1.42
23	1.58	1.98
24	1.76	2.42
25	1.79	2.91
26	1.85	3.38
27	1.96	4.44
28	2.23	5.61
29	2.72	7.05
30	3.13	9.14

C DIFFERENCE OUTPUT FROM CORONET WARRIOR DATA

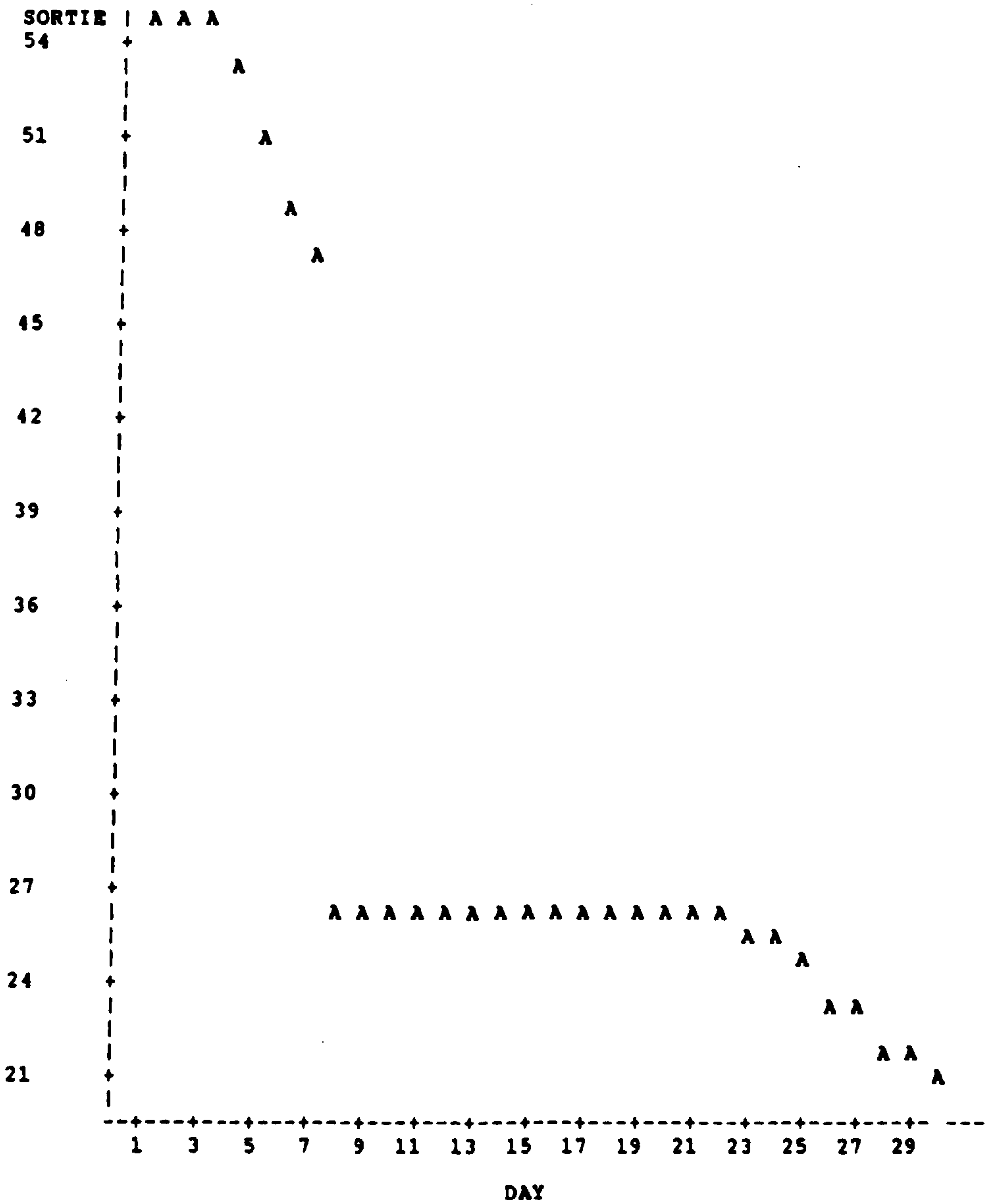
DAY	FMC	SORTIES
1	1.04	-0.20
2	1.75	-0.15
3	0.59	-0.19
4	0.35	-0.18
5	0.21	-0.16
6	0.16	-0.12
7	0.26	-0.13
8	0.08	-0.40
9	0.17	-0.40
10	0.29	-0.40
11	0.40	-0.40
12	0.44	-0.40
13	0.42	-0.40
14	0.39	-0.40
15	0.40	-0.40
16	0.41	-0.40
17	0.38	-0.40
18	0.29	-0.40
19	0.20	-0.40
20	0.20	-0.40
21	0.25	-0.40
22	0.29	-0.40
23	0.30	-0.40
24	0.30	-0.40
25	0.29	-0.40
26	0.21	-0.40
27	0.31	-0.40
28	0.49	-0.39
29	0.50	-0.39
30	0.63	-0.38

```
C
C   CALCULATED THE MEANS AND STANDARD DEVIATIONS FOR
C   THE COMPUTED DIFFERENCES BETWEEN THE RESEARCH
C   MODEL AND THE DYNA-METRIC MODEL
C   (SAS PROGRAM)
C

options ls=80;
filename new'DIFF.dat';
data new;
infile new;
input DIFF;
PROC MEANS;
```

```
C
C   PROGRAM USED TO PLOT OUTPUT DATA
C   (SAS PROGRAM)
C

options ls=80;
filename new'TEST3A.DAT';
data new;
infile new;
input DAY FMC SORTIE;
PROC PLOT;
 PLOT SORTIE*DAY FMC*DAY;
```

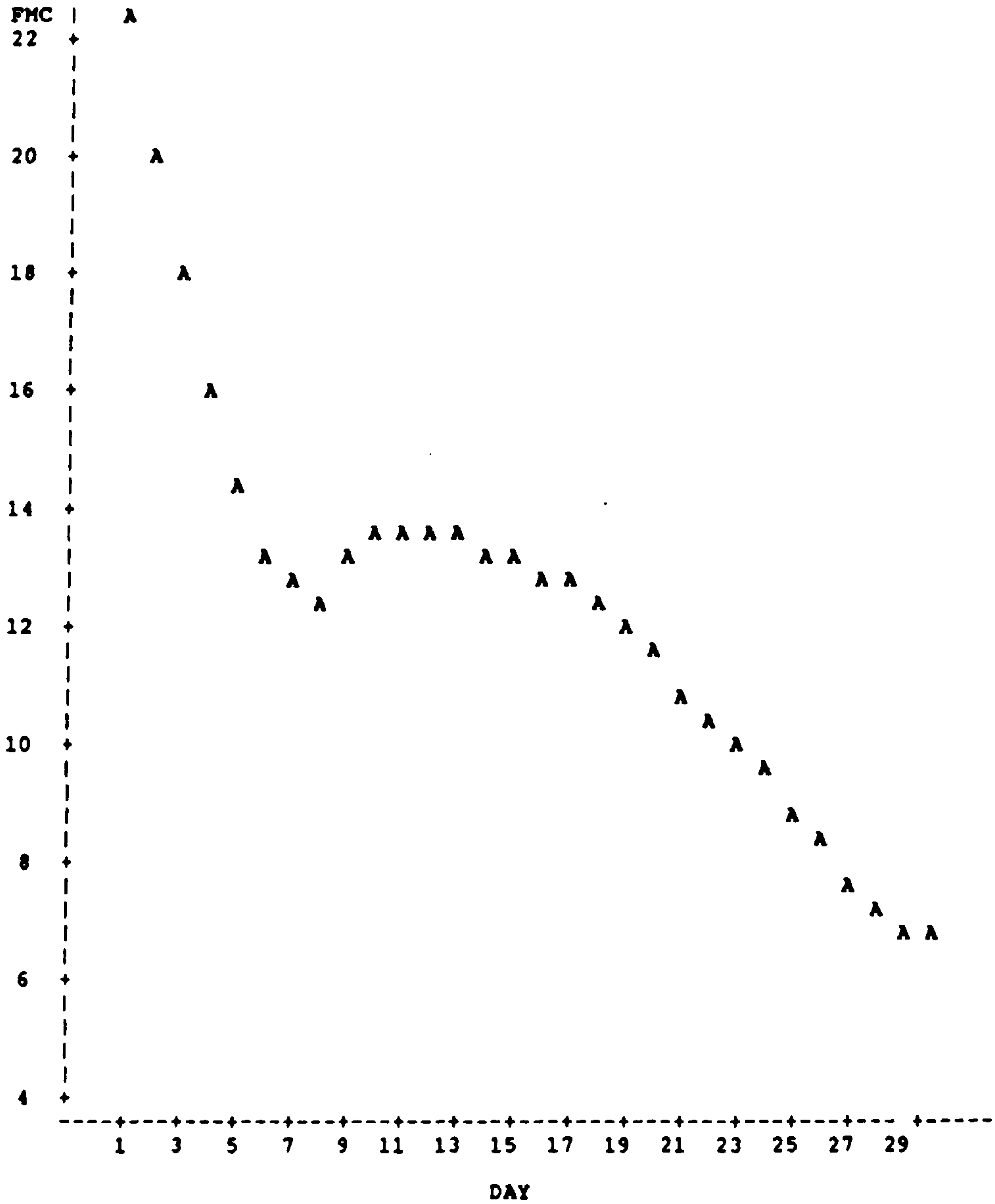
RESEARCH MODEL OUTPUT TAC SAMPLE DATA

PLOT OF SORTIE*DAY   LEGEND: A = 1 OBS, B = 2 OBS, ETC.



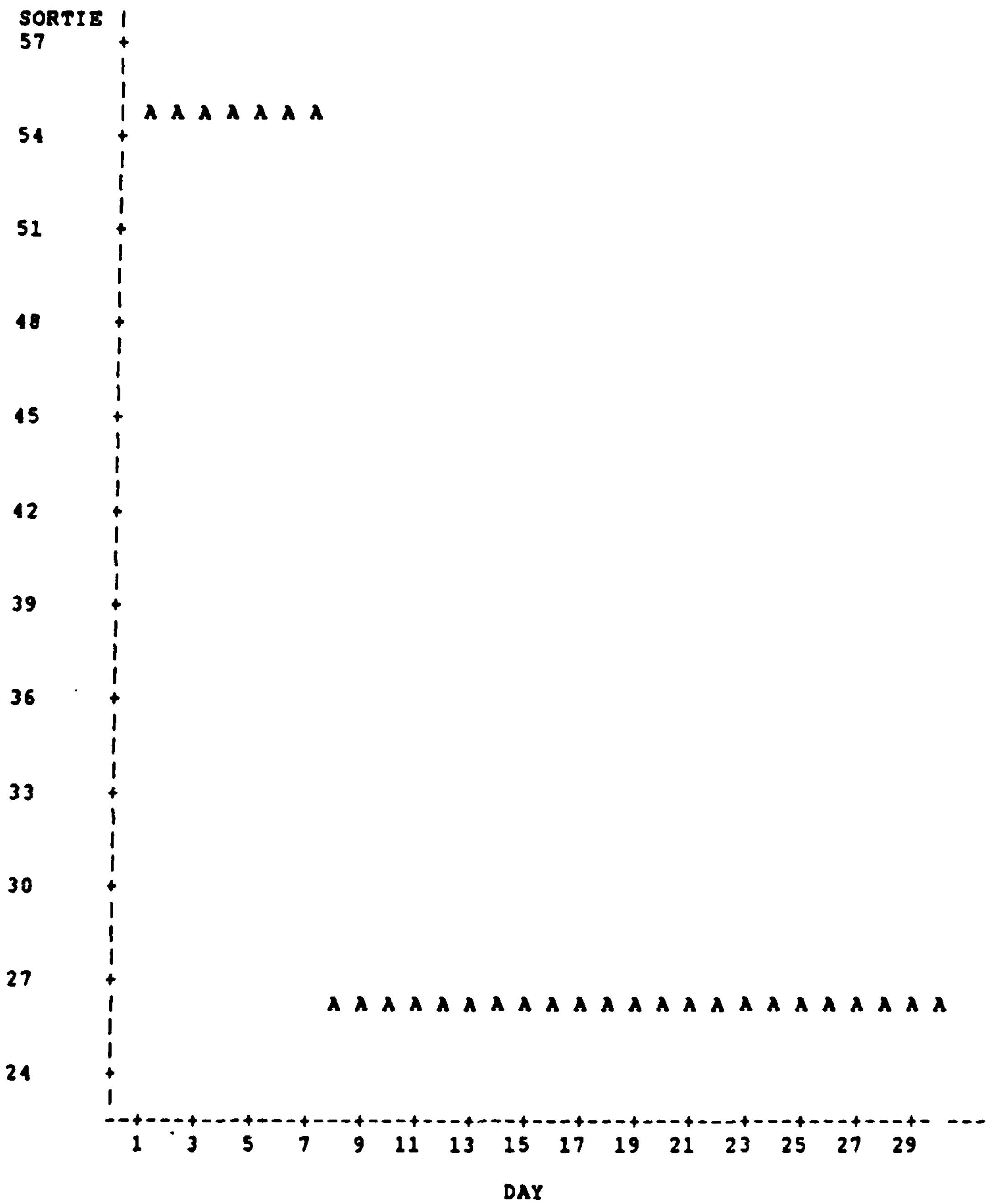
RESEARCH MODEL OUTPUT TAC SAMPLE DATA

PLOT OF FMC*DAY   LEGEND: A = 1 OBS, B = 2 OBS, ETC.



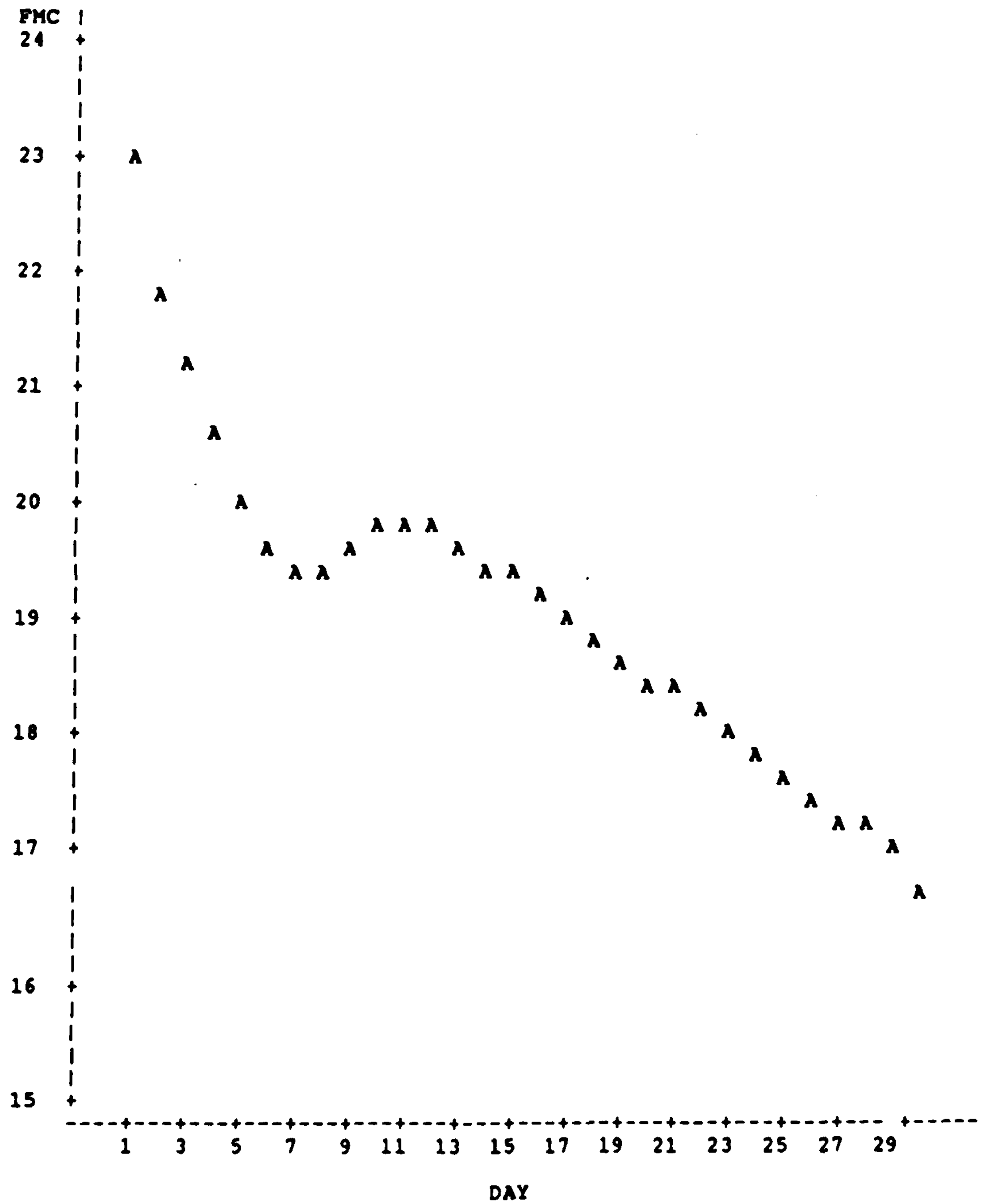
RESEARCH MODEL OUTPUT CORONET WARRIOR DATA

PLOT OF SORTIE*DAY   LEGEND: A = 1 OBS, B = 2 OBS, ETC.



RESEARCH MODEL OUTPUT CORONET WARRIOR DATA

PLOT OF FMC*DAY   LEGEND: A = 1 OBS, B = 2 OBS, ETC.



BIBLIOGRAPHY

1. Aircraft Subsystem Availability Model (ASAM). Capt. Bob O'Neill, Talking paper. HQ AFOTEC/LG: Kirtland AFB, N.M., March 1986.

2. An Expected-Value-Based Logistics Capability Assessment Model (ELCAM). Major Roy E. Rice, Talking Paper. Air Force Logistics Management Center (AFLMC/LGY), Gunter AFS, AL.

3. Arthur, Jeffrey L., James O. Frendewey, Parviz Ghandforoush and Loren Paul Rees. "Microcomputer Simulation Systems," Computers and Operations Research 13: 167-183 (1986).

4. BDM Corporation. Availability/Readiness Model For The Personal Computer (ARM-PC): Copy available from BDM.

5. Budde, Michael J., Chief, Logistics Analysis Branch. Personal interviews. HQ TAC/LGY, Langley AFB, Va., April 28 through September 30, 1987.

6. Conover, W. J. Practical Nonparametric Statistics (Second Edition). New York, New York: John Wiley and Sons, 1980.

7. CORONET WARRIOR. Briefing by Maj. Gen. Henry Viccellio, Jr. HQ TAC/LG, Langley AFB, Va., August 5, 1987. Slides available from author.

8. Dalkey, Norman C. "Simulation," Systems Analysis and Policy Planning: Applications in Defense: 241-254 (June 1968).

9. Demmy, W. Steven and Jon R. Hobbs. "Dyna-METRIC: an Overview," Air Force Journal of Logistics: 14-17 (Spring 1983).

10. Engelbeck, Capt Ranald M. and Capt Roland K. Lough. The War Readiness Spares Kit: An Examination of the Concept, Interpretation, and Kit Utilization by Tactical Fighter Forces. MS Thesis, AFIT/LM/SLSR-11-66. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, August 1966 (AD-805632).

11. Friedman, Linda W. and Hershey H. Friedman. "Validating the simulation metamodel: Some practical approaches," Simulation, 45: 144-146 (September 1985).

12. Gage, Thomas W. and Capt. Andrew J. Ogan, USAF. "Dyna-METRIC: A Valuable Management Tool," Air Force Journal of Logistics: 22-24 (Spring 1983).

13. Glazener, Capt Charles U. and Capt Alan J. Tinder. War Readiness Spares Kit Composition: A Systems Approach. MS Thesis, AFIT/LM/SLSR-30-76B. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1976 (AD-A032426).

14. Hawthorne, Daniel. For the Want of a Nail: The Influence of Logistics on War. New York: Whittlesay House, McGraw Hill Co., Inc., 1948.

15. Headquarters TAC/LGY Sponsored Thesis Project: Microcomputer Sortie Generation Simulation System. Capt. Michael J. Budde, Talking paper. HQ. TAC/LGY, Langley AFB, Va., April 28, 1987.

16. Hillestad, R. J. and M. J. Carrillo. Models And Techniques For Recoverable Item Stockage When Demand And The Repair Process Are Nonstationary--Part 1: Performance Measurement. Contract N-1482-AF. Santa Monica Ca: Rand Corp., May 1980.

17. Hillier, Frederick S., Gerald J. Lieberman. Introduction to Operations Research (Fourth Edition). Oakland, California: Holden-Day,Inc., 1986.

18. Lee, Sang M., Laurence J. Moore, and Bernard W. Taylor. Management Science. Dubuque, Iowa: Wm. C. Brown Company, 1981.

19. Litko, Joseph R. OPER 666 SIMULATION I (Class Notes). AFIT School of Engineering, Jan. 1987.

20. Mabe, Richard D. LOG 290 Combat Capability Assessment Class Notes. AFIT School of Systems and Logistics Management, June 1987.

21. -----. Instructor, AFIT School of Systems and Logistics Management. Personal interviews. AFIT, Wright Patterson AFB, OH., August 6 through September 4, 1987.

22. O'Keefe, Robert and Ruth Davies. "A microcomputer

system for simulation modeling," European Journal of Operations Research, 24: 23-29 (January 1986).

23. Pritsker, A. Alan B. Introduction to Simulation and SLAM II (Third Edition). West Lafayette, Ind.: Systems Publishing Corp., 1986.

24. Pyles, Raymond. The Dyna-METRIC Readiness Assessment Model Motivation, Capabilities, and Use. Contract F49620-82-C-0018. Santa Monica Ca: Rand Corp., July 1984.

25. -----, Raymond and Lt. Col. Robert S. Tripp, USAF. "Measuring and Managing Readiness: An Old Problem- A New Approach," Air Force Journal of Logistics: 18-21,24 (Spring 1983).

26. Quade, E. S. "Pitfalls And Limitations," Systems Analysis and Policy Planning: Applications in Defense: 345-363 (June 1968).

27. The Ryrie Study Bible. Charles Caldwell Ryrie, editor. Chicago, Ill.: Moody Press, 1978.

28. Shannon, Robert E. Systems Simulation the art and the science. Englewood Cliffs, N.J.: Prentice Hall, Inc., 1975.

29. Specht, R. D. "The Nature Of Models," Systems Analysis and Policy Planning: Applications in Defense: 211-227 (June 1968).

30. TAC PACERS II. Briefing by Maj. Duane Murphy. HQ TAC/LGY, Langley AFB, Va., April 27, 1987. Slides available from author.

31. Torn, Aimo A. "Simulation nets, a simulation modeling and validation tool," Simulation, 45: 71-75 (August 1985).

32. U.S. Department of the Air Force. U.S. Air Force Supply Manual. AFM 67-1 Vol. 1 Part 1 Chap. 14, Washington D.C., Sept. 29, 1986.

33. -----. U.S. Air Force Supply Manual. AFM 67-1, Vol. II, Part II, Chap. 20 Tac Supplement (Draft). Langley AFB, VA: HQ TAC, Oct. 1987.

34. -----. U.S. Department of the Air Force Logistics Long-Range Planning Guide FY 1987-2001. DCS/Logistics and Engineering, Headquarters USAF, Washington D.C., Sept. 10, 1984.

35. -----. WRM Program. AFR 400-24 Chap.1, Washington D.C., Oct. 24, 1983.

36. -----. War Reserve Materiel Compendium: Users Guide Vol. II. WPAFB, Ohio: 2750 LS/DMSPT.

37. Vasudev, Vinay K. and William E. Biles. "Microcomputer-based Modeling and Simulation," European Journal of Operations Research, 24: 30-36 (January 1986).

38. Wackerly, Dennis D., Richard L. Scheaffer, and William Mendenhall. Mathematical Statistics with Applications (Third Edition). Boston: Duxbury Press, 1981.

COMPUTER PACKAGES

39. AID. West Lafayette, Ind.: Pritsker and Associates, Inc., 1981.

40. SAS. Cary, N.C.: SAS Institute Inc., 1986.

41. SLAM II PC. West Lafayette, Ind.: Pritsker and Associates, Inc., March 1986.

VITA

Captain Theodore P. Lewis was born on March 2, 1961 in Butte, Montana. He graduated from Great Falls High school in Great Falls, Montana. He then went to the United States Air Force Academy where he graduated in 1983 with a Bachelor of Science. After graduation, he was assigned to the Air Force Operational Test and Evaluation Center in Albuquerque, New Mexico. While there he worked in the Avionics and Electronics Logistics Analysis Branch. From there he went to the Air Force Institue of Technology, WPAFB, Ohio in 1986.

Permanent address: 95 Blue Wing
Walkerville, Mont. 59701

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GOR/ENS/87D-9	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION SCHOOL OF ENGINEERING	6b. OFFICE SYMBOL *(If applicable)* AFIT/ENS
7a. NAME OF MONITORING ORGANIZATION	
6c. ADDRESS *(City, State, and ZIP Code)* AIR FORCE INSTITUTE OF TECHNOLOGY WRIGHT PATTERSON AFB, OH 45433-6583	7b. ADDRESS *(City, State, and ZIP Code)*
8a. NAME OF FUNDING/SPONSORING ORGANIZATION	8b. OFFICE SYMBOL *(If applicable)*
9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c. ADDRESS *(City, State, and ZIP Code)*	10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*
SEE BOX 19

12. PERSONAL AUTHOR(S)
THEODORE P. LEWIS, B.S., CAPT., USAF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
MS THESIS	FROM ______ TO ______	1987 DECEMBER	153

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP
15	05	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
SIMULATION, SPARE PARTS, LOGISTICS PLANNING

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

UNIT LEVEL WRSK ASSESSMENT AND SORTIE
GENERATION SIMULATION MODEL

THESIS CHAIRMAN: PHILLIP D. MILLER, MAJOR, USAF
ASSOCIATE PROFFESSOR OF LOGISTICS MGT.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE *(Include Area Code)*	22c. OFFICE SYMBOL
PHILLIP D. MILLER, PHD, MAJOR, USAF	1-513-255-5023	AFIT/LGM

Abstract

HQ TAC/LGY uses Dyna-METRIC as a WRSK assessment tool but they have expressed a need for a more flexible model that is capable of running on a microcomputer. For example, Dyna-METRIC has a number of limiting assumptions such as assuming unlimited maintenance capacity. The purpose of this thesis work was to develop a model to emulate and extend the Dyna-METRIC modeling capability.

To begin this research a simulation package had to be chosen. Microcomputer simulation languages were compared and SLAM II PC was selected because of its price, portability, widespread acceptance as a simulation language, and the availability of the software.

Another area of concern was Dyna-METRIC's use of the exponential distribution to model repair times. Questions have arisen as to whether this is a reasonable assumption or whether the lognormal distribution provides a better fit. The sample repair times were taken from a TAC exercise called Coronet Warrior. The results were inconclusive due, primarily, to the small sample sizes. Testing of the research model centered around two data sets. The first was provided by HQ TAC/LGY and the second came from the TAC Coronet Warrior exercise. The outputs of interest are sorties per day and number of fully mission-capable aircraft available per day. Each data set was used with the research and Dyna-METRIC models. The outputs were then compared by day and type. A hypothesis test of the differences was performed. The differences were not found to be statistically different from zero. Therefore, the research model provides a reasonable emulation of the Dyna-METRIC model with respect to the outputs of interest. Future research is recommended in input and output formats and in variance reduction techniques to reduce the number of simulation runs necessary.

END
DATE
3-88
DTIC